

LIBRARY USE ONLY

0262

UNCLASSIFIED

NAVAL UNDERWATER SYSTEMS CENTER
NEWPORT LABORATORY
NEWPORT RHODE ISLAND 02841-5047
RETURN TO: TECHNICAL LIBRARY

NUWC-NPT TM 922000

NAVAL UNDERSEA WARFARE CENTER DIVISION
NEWPORT, RHODE ISLAND



Technical Memorandum

DESIGN, OPERATION, AND EVALUATION
OF THE NUWC TRANSIENT IMPELLER TEST FACILITY

Date: 2 January 1992

Robert G. Gregory
Robert G. Gregory

Paul J. Lefebvre
Paul J. Lefebvre

William P. Barker
William P. Barker

LIBRARY USE ONLY

Mark Rodriguez
Mark Rodriguez

Launcher Systems
Development Division

Launcher & Missile
Systems Department

Approved for public release; distribution is unlimited.

Report Documentation Page			Form Approved OMB No. 0704-0188		
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 02 JAN 1992		2. REPORT TYPE Technical Memorandum		3. DATES COVERED 02-01-1992 to 02-01-1992	
4. TITLE AND SUBTITLE Design, Operation, and Evaluation of the NUWC Transient Impeller Test Facility			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Robert Gregory; Paul Lefebvre; William Barker; Mark Rodrigues			5d. PROJECT NUMBER A64953; J44100		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Undersea Warfare Center Division, Newport, RI, 02841			8. PERFORMING ORGANIZATION REPORT NUMBER TM 922000		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES NUWC2015					
14. ABSTRACT A new facility, the Naval Undersea Warfare Center's Impeller Test Facility, has been designed and built for the purpose of conducting research and development into the hydrodynamics and hydroacoustics of transient operation of pumps. The facility's capabilities include steady-state operation and user-defined transient operation through simultaneous control of pump rotational speed and flowrate. Transient operation of 1/3 scale submarine weapon ejection pumps can be accurately modelled for investigations of hydrodynamic performance, cavitation, hydroacoustic noise sources, and details of the flow field. Design and operational procedures are provided in detail in this report.					
15. SUBJECT TERMS Low Noise TPES Impeller; Transient Impeller Technology; pumps					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 88	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

ABSTRACT

A new facility, the Naval Undersea Warfare Center's Impeller Test Facility, has been designed and built for the purpose of conducting research and development into the hydrodynamics and hydroacoustics of transient operation of pumps. The facility's capabilities include steady-state operation and user-defined transient operation through simultaneous control of pump rotational speed and flowrate. Transient operation of 1/3 scale submarine weapon ejection pumps can be accurately modelled for investigations of hydrodynamic performance, cavitation, hydroacoustic noise sources, and details of the flow field. Design and operational procedures are provided in detail in this report.

ADMINISTRATIVE INFORMATION

This memorandum was prepared under the joint sponsorship of two projects. The first was Project No. A64953, "Low Noise TPES Impeller," principal investigator P. J. Lefebvre (Code 8322), sponsored by the Naval Undersea Warfare Center's Independent Research Program.

The second was Project No. J44100, "Transient Impeller Technology," principal investigator P. J. Lefebvre (Code 8322), sponsored by the Office of Naval Technology under the Advanced Launchers Project of the Submarine Technology Block Program.

The authors of this memorandum are located at the Naval Undersea Warfare Center Division, Newport, RI 02841-5047.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS	iv
NOMENCLATURE	iv
1. INTRODUCTION	1
2. FACILITY HARDWARE AND CONTROL SYSTEMS DESCRIPTION	2
3. INSTRUMENTATION	11
4. FACILITY EVALUATION	14
5. OPERATIONAL PROCEDURES	15
GENERAL	15
STEADY STATE	15
TRANSIENT	19
6. DATA ACQUISITION, REDUCTION, AND PRESENTATION	20
GENERAL	20
DATA ACQUISITION	20
MASSCOMP DATA PRESENTATION	22
DATA REDUCTION	23
PERFORMANCE DATA PRESENTATION	25
7. REFERENCES	26
APPENDIX A: CONTROL SYSTEMS SOURCE CODE	A-1
APPENDIX B: MOTOR CONTROLLER OPERATION	B-1
APPENDIX C: DEAERATION AND FILTRATION PROCEDURES	C-1
APPENDIX D: FACILITY OPERATION	D-1
APPENDIX E: DATA ACQUISITION, REDUCTION, AND PRESENTATION FLOWCHART	E-1
APPENDIX F: LASER SAFETY PROCEDURES FOR BUILDING 1246	F-1

LIST OF ILLUSTRATIONS

2.1. IMPELLER TEST FACILITY LAYOUT	3
2.2. IMPELLER TEST FACILITY	4
2.3. TEST SECTION	5
2.4. IMPELLER HOUSING	6
2.5. DC DRIVE MOTORS	7
3.1. INSTRUMENTATION LOCATION	12
4.1. RPM & FLOWRATE VS TIME	16
4.2. C_q VS TIME	17
4.3. C_h VS C_q	18

NOMENCLATURE

C_h static head coefficient

C_q flow coefficient

N rotational speed (RPM)

Q flow rate (gal/min)

RPM rotational speed (RPM) (used interchangeably with N)

1. INTRODUCTION

Although pumps (impellers) have been used for many years as prime movers in hydraulic systems, most of these applications were for steady-state operation where pump performance at startup or down was of no concern. Several applications in the Navy and industry however require controlled transient operation from startup to shutdown. One such Navy application is the torpedo ejection pump impeller used to launch weapons from submarines. This impeller operates under highly transient conditions throughout the launch.

To date, torpedo ejection pump impellers, along with those for other transient applications, have been designed exclusively by applying quasi-steady analysis and techniques and ignoring any potential transient effects. This amounted to assuming that at any instant in time during a transient, all flow conditions and, hence, pump performance was identical to that which existed during steady-state operation while operating at the instantaneous pump speed and flow rate. This approach was necessary since little was known about the physics of transient impeller operation.

Increasingly demanding hydrodynamic and acoustic noise performance requirements for torpedo ejection pumps were imposed as noise goals for SSN 21, and follow-on SSN launch systems became significantly more stringent. A better technical understanding of transient impeller operation was then required in order to develop successful cavitation-free, low noise design methodologies. Improved understanding of the underlying transient physics of the impeller flow field is critical if the noise sources created by the pump are to be identified and subsequently attenuated or eliminated. Future launch impeller designs must be based on the actual physics which prevail during the transient operation.

There have been many research efforts in the past to define the flow field and identify hydroacoustic sources in rotating turbomachinery operating under steady-state conditions. However, even though fairly sophisticated experimental and computational tools have been developed, much further development is still required to fully understand the complex phenomena in these relatively simple flows (relative to the transient case). Only some of these techniques are applicable to transient pump operation, where up until the present time, very few experimental and numerical studies have been undertaken. New experimental and numerical techniques for transient operation are obviously required.

Developing the necessary transient technology can only be accomplished by testing candidate impellers under both steady-state and transient conditions. This goal can be realized with the Naval Undersea Warfare Center's (NUWC) Impeller Test Facility (ITF). This facility is a unique test bed for the study and development of transient technology. The ability to interchange different impeller designs, coupled with the ability to accurately model transient impeller operation makes this facility an

invaluable tool for the study of impeller hydrodynamics and hydro-acoustics. Also, since the facility can operate under both steady-state and transient conditions, the quasi-steady assumptions used to date in pump impeller design can be evaluated and confidence can be found in the results. Experiments can also now be conducted to investigate details of the hydrodynamic and hydro-acoustic phenomena under both operating conditions.

This report describes the test facility hardware, control systems, instrumentation, operational procedures, data acquisition, data reduction, and data presentation of the Impeller Test Facility.

2. FACILITY HARDWARE AND CONTROL SYSTEMS DESCRIPTION

The Naval Undersea Warfare Center's ITF is located in Building 1246 of the Newport, Rhode Island Laboratory. As previously stated, it was designed and built to support research, development, and engineering studies investigating basic impeller hydrodynamic and hydro-acoustic phenomena under both steady-state and transient flow conditions.

Unlike other facilities which provide only steady-state or transient operation, this facility can provide both operational modes and has user defined transient control not available on any other known facility. Transient operation control systems have been developed which operate simultaneously during transient operation in order to produce user specified impeller rotational speed (N) and flow rate (Q) time histories as required to properly model full scale conditions. The facility is versatile and conducive to detailed instrumented and flow visualization experiments that can be conducted to aid in the analysis of the overall performance of existing impellers, allow research into the underlying physics of the transient impeller flow field, and provide a tool for the design and optimization of future impellers.

The overall facility layout is shown in Figure 2-1. Photographs of the facility are shown in Figures 2-2 through 2-5. The facility is a closed loop re-circulating system and is capable of being pressurized up to 110 psig. Major components are an impeller test section, a 16-inch programmable control valve, a 16-inch manual loading valve, a settling tank containing honeycomb flow/turbulence management units and discharge nozzle, a NUWC-developed transient flow meter, and two 200 hp low inertia DC motors to drive the impeller.

As shown in Figure 2-1, the impeller under test is housed in the test section. The test section was designed to simulate the near impeller flow field of an actual submarine impulse tank. The design of the impulse tank is versatile so as to accommodate a wide range of impellers and the addition of a diffuser or collector at the impeller discharge if required in the future.

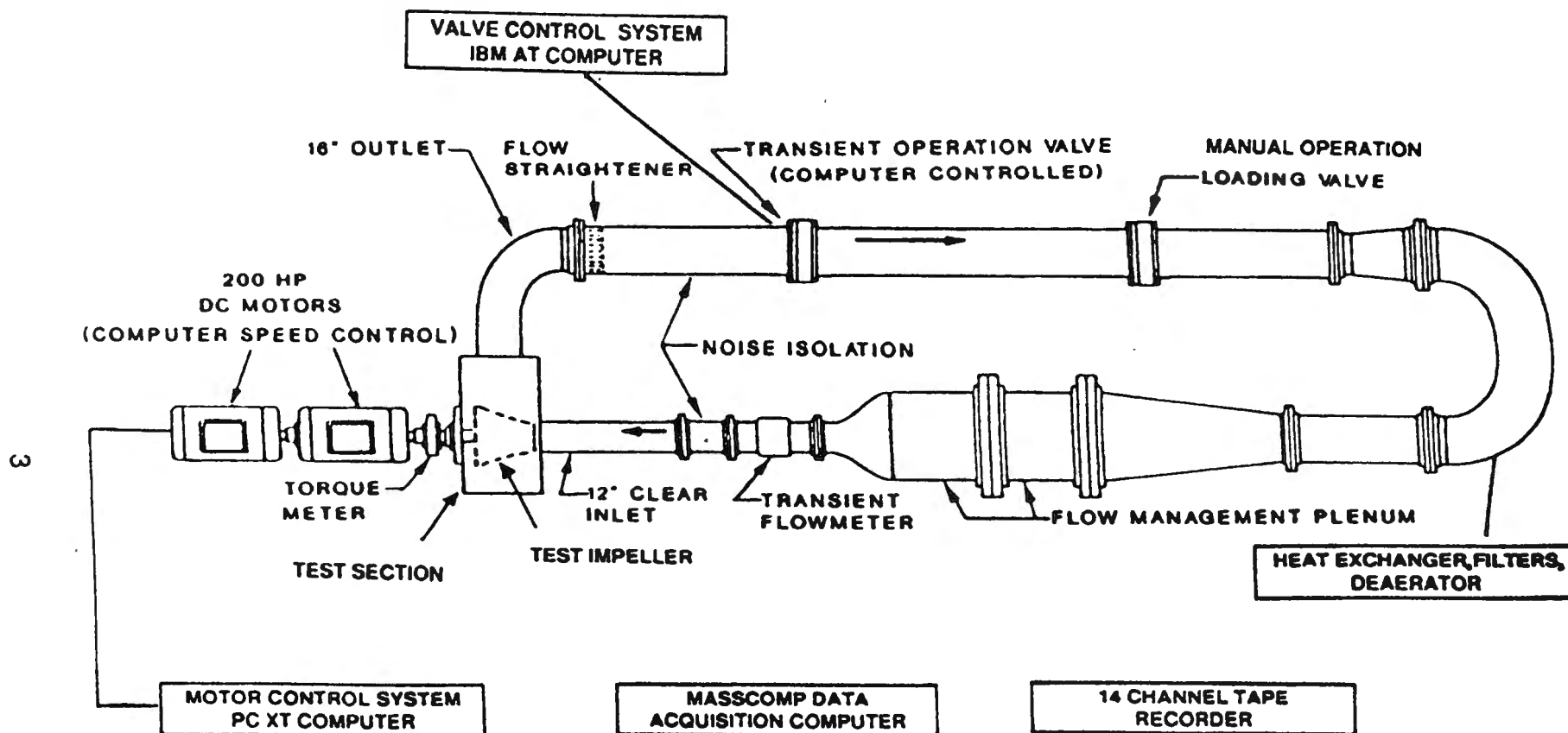


Figure 2-1. Test Facility Layout

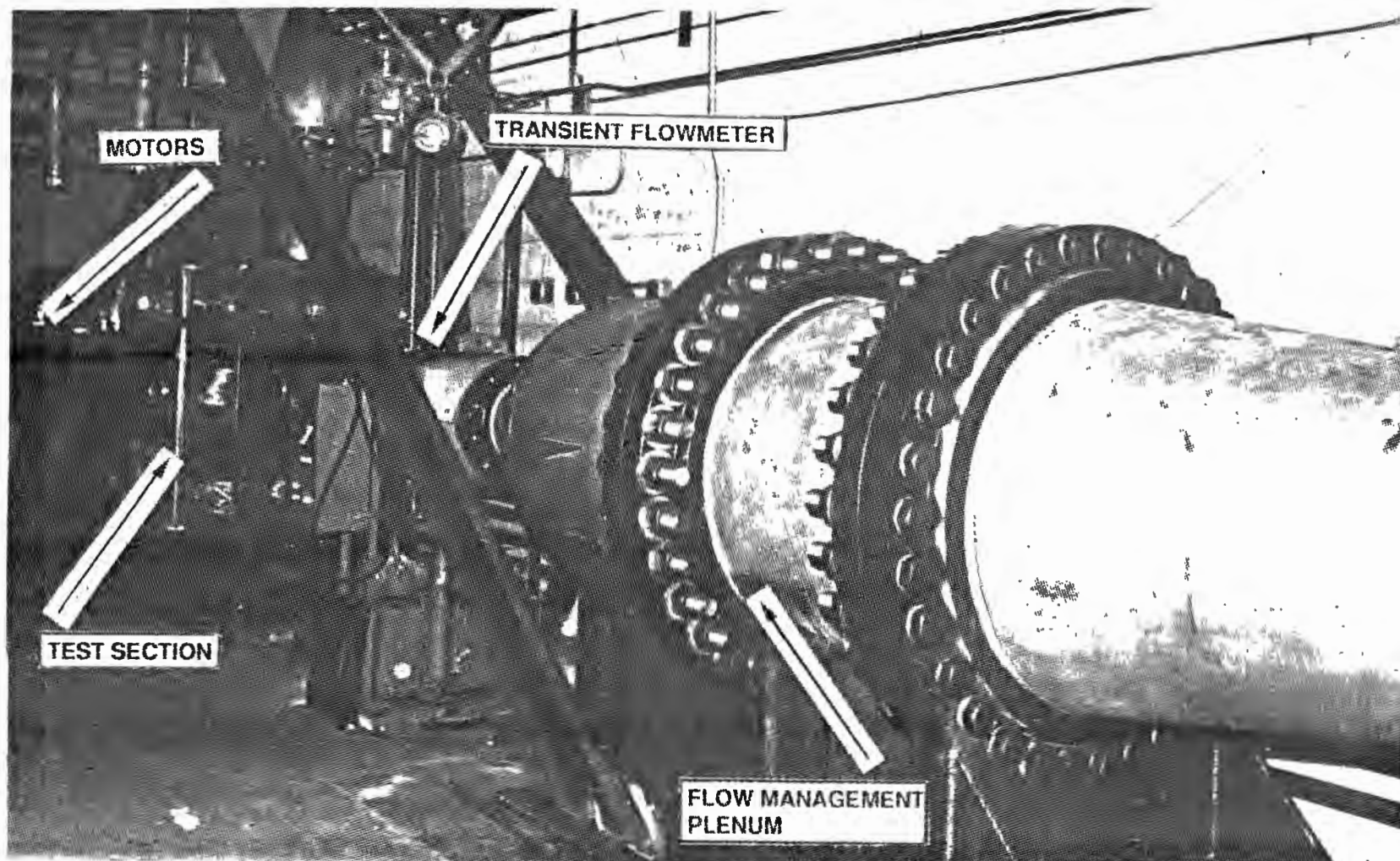


Figure 2-2. Impeller Test Facility

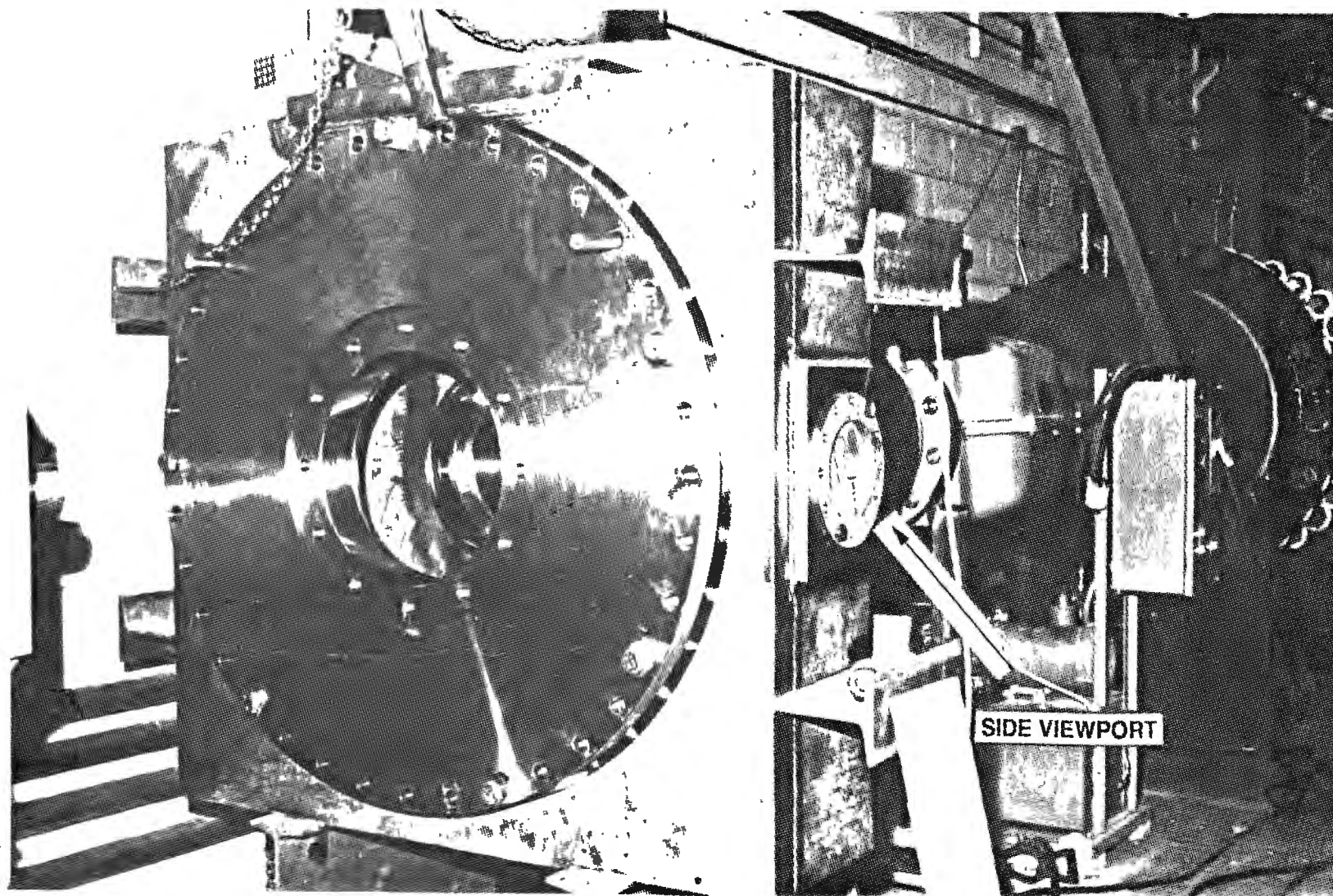


Figure 2-3. Test Section

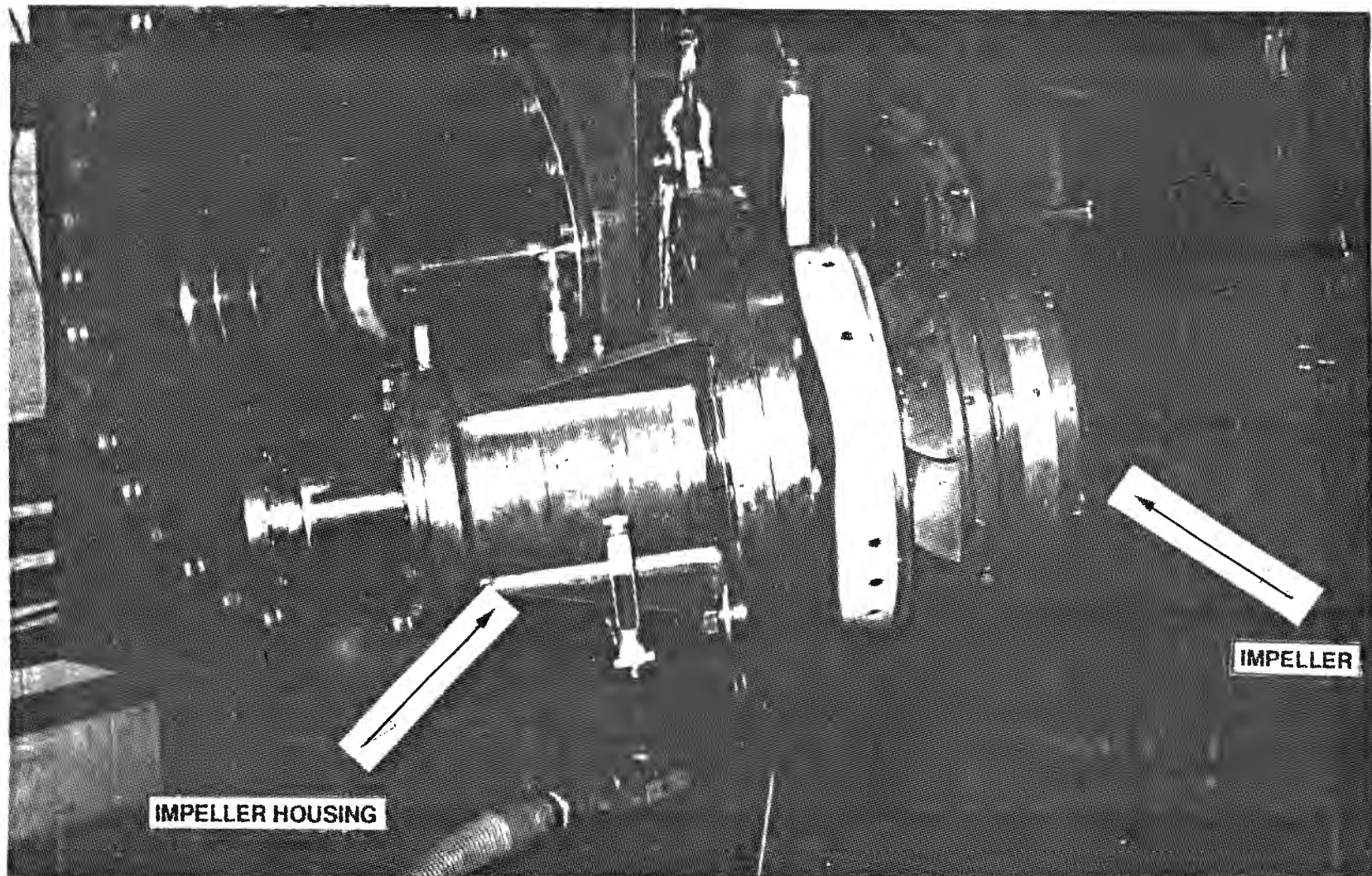


Figure 2-4. Impeller Housing

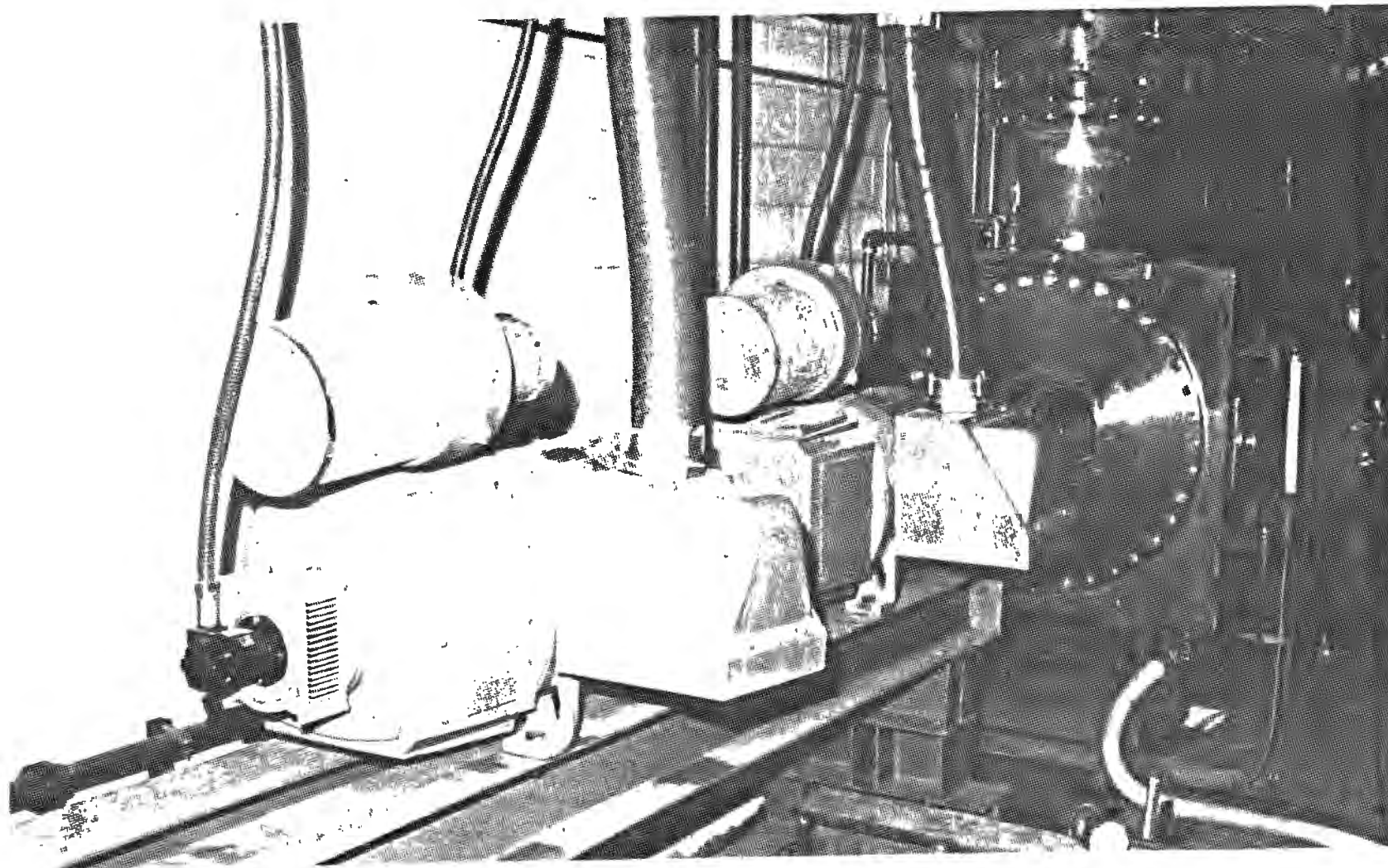


Figure 2-5. DC Drive Motors

Visual access for flow visualization and laser Doppler velocimeter (LDV) measurements at the impeller discharge is accommodated by a 6-inch-diameter view port that was included at each side of the test section. The inlet flow to the impeller is through a 12-inch-diameter clear cast acrylic pipe which allows visual access to the blades' leading edge. Inception of leading edge cavitation can be monitored through this acrylic pipe and correlated with the acoustic sensor data.

The flow in the facility is developed by the test impeller which is powered by two General Electric 200 hp low inertia DC electric motors connected in series. The impeller under test is directly coupled to the motors so that the impeller rotates at the same speed as the motors. A torque meter is mounted between the motors and the impeller. The DC motors provide sufficient power and torque to overcome the inertia of the motors and to accelerate the flow as required to accurately model full scale transient launch conditions.

The instantaneous rotational speed of the motors, N (RPM), is controlled by a dedicated microprocessor based feedback control system which has both steady-state and transient operating capabilities.

The controller is a Mentor 55R dedicated microprocessor from Burton Industries. It is housed in the electrical cabinet located at the ITF which also contains the miscellaneous electrical hardware such as silicon controlled rectifier (SCR) switches and fuses. The Mentor 55R controls both steady-state and transient motor operation and all of the operating parameters including those affecting control, safety, and user defined limits. The Mentor parameters can be set via an integrated key pad, but since it is only accessible with the cabinet doors open and with the power interrupted by the door interlock, it is only useful to a certified electrician who can bypass the interlock during repair. An IBM compatible PC XT is therefore used for all normal interfacing with the controller, which includes the setting of all safety and operational parameters. The PC XT communicates with the Mentor through an RS-232 serial communications port.

The feedback signal is the RPM from a tachometer mounted on the motor shaft at the rear of the motors. The command signal, whether one constant value of RPM for steady-state operation or a file of RPM versus time for transient operation, is downloaded from the PC directly to the Mentor. No adjustments to the transient RPM versus time file are required as long as the correct gain factors for proportional and integral gain are set, resulting in a fully automatic system. The Mentor can accommodate updated RPM commands at time intervals as low as 0.015 seconds. However, 0.025 seconds is adequate for typical weapon launch simulation. Steady-state speed control is also provided through a potentiometer mounted near the electrical cabinet.

The computer code for interfacing the PC to the motor controller is **MOTORS.BAS**, a program written in the BASIC programming language. This program allows test personnel to perform the aforementioned tasks including downloading of the command signal, whether it be a steady-state constant RPM value or a file of RPM versus time for transient operation of the facility. For transient operation, this program allows the test personnel to put the motor controller into a wait state. The PC XT

remains in this wait state until the control character, "A", is sent from a second PC, an IBM AT, that serves as a trigger for test initiation. (The complete function of the IBM AT personal computer is given later.) Upon receiving the trigger signal, the XT sends a signal to the Mentor 55R microprocessor to initiate the motor transient. Updated RPM values are sent to the Mentor at the predetermined time step increments. Once the final RPM value is sent to the Mentor, MOTORS.BAS, as currently set up, disables the thyristers (parameter #166 = 1) to allow the motors to coast down. A listing of the source code for MOTORS.BAS is contained in Appendix A.

The time step associated with each new RPM value during transient operation is also set in MOTORS.BAS. This is accomplished by providing the proper number of iterations through the DO loop prior to sending the next RPM value to the Mentor. The number of DO loop iterations is determined by examining the command signal output on a storage oscilloscope. Procedures for running the motor controller are found in Appendix B. A partial listing of the commonly used motor controller parameters are as follows:

#155 = 0	Motor RPM via Potentiometer
= 1	Motor RPM via Computer
#166 = 0	Enable Thyristers
= 1	Disables Thyristers
#55 = 0 to 255	Proportional Gain
#56 = 0 to 255	Integral Gain

When using computer control:

#22 = No. of counts x 2.4024 (RPM/counts) Motor RPM

Flow rate is controlled by a second control system which regulates the motion of the 16-inch-diameter control valve installed downstream of the test section. The control system can be operated in a steady-state mode or provide a transient valve position time history and hence transient flow rate control. The feedback signal is the valve position as measured by a position indicator mounted on the valve stem, and the command signal is generated by the same IBM AT personal computer used to trigger the XT. This system has the same design as the control system used for the NUWC Unsteady Flow Loop Facility's valve control, as described in reference [1] and in more detail in the following paragraph.

In essence, the IBM AT, with the VCONIMPL.BAS program (source listing provided in Appendix A), is the backbone of the facility control system. It controls the initiation of all events including gating of data acquisition hardware and triggering of the motor control system, as mentioned above. The triggering of the control and data acquisition system is as follows. First, an Omega Engineering digital to analog (DAC) board converts a binary signal from the IBM AT into an analog voltage. This analog voltage is a square wave, low to high voltage signal, which is sent to the MASSCOMP computer, initiating data acquisition. This trigger signal can also be sent to a tape recorder, if it is being used for any of the data channels. Following a 50-millisecond

delay, which is set in line 235 of the VCONIMPL.BAS program, the RS 232 serial communication board located at communications port 2 of the AT machine sends the letter "A" as a trigger signal to the PC XT computer to initiate motor startup. The IBM AT then begins to move the control valve through the use of the second channel of the Omega DAC board, which converts the digital valve position command signal from VCONIMPL.BAS into analog output voltage. This voltage signal is sent to the Moog analog controller board. The valve position feedback signal is also sent to this board, and the error signal resulting from the difference between the feedback and command signals controls the current to the Moog servo valve. This servo valve, which is connected to a 3000 psi, 1.0 gpm Double A hydraulic pump, controls the flow of hydraulic fluid to the actuator on the control valve.

The computer program that allows the IBM AT to perform its tasks is the aforementioned VCONIMPL.BAS, written in the BASIC programming language. This code allows for both the steady-state and transient operation of the aforementioned 16-inch control valve. In the case of steady-state operation the program allows test personnel to move the control valve to any desired position from full closed (0 degrees) to full open (90 degrees). For transient operation a data file containing valve position versus time is loaded for valve operation. The valve position is updated at 0.01 second time intervals. Unlike the motor control system, the flow rate is not controlled directly, thus valve position versus time data must be determined by a trial and error method. This method was facilitated through the use of a computer simulation of the facility's transient operation, which allowed the valve position data file to be determined with very few iterations.

The programs VCONIMPL.BAS and MOTOR.BAS combine to produce the aforementioned capability to simultaneously control the impeller rotational speed, N , and the volumetric flowrate, Q . Thus, the control systems allow for the accurate modeling of any transient event desired.

Between the control valve and the test section is a 10-foot-long section of 16-inch ID rubber pipe to attenuate structureborne vibrations and fluidborne noise. A second 16-inch-diameter valve is installed downstream of the control valve. This is a manually actuated valve and is used to "load" or add additional hydrodynamic losses in the facility. In this way, the pressure drop required in the facility to control flow rate is accomplished across both the manual and the transient valve resulting in reduced noise and a lower susceptibility to cavitate than if the total pressure drop occurred over just one valve. During a transient run, the manual valve is set to a stationary position which has been chosen to provide an optimum distribution of pressure drop across each of the valves.

The flow into the 12-inch-diameter clear acrylic inlet pipe is conditioned by a 36-inch diameter settling tank and nozzle which is of the same design as that in the NUWC Unsteady Flow Loop Facility that is described fully in reference [2]. The 36-inch-diameter of the settling tank provides a 9:1 area reduction into the inlet pipe. Stainless steel honeycomb is sandwiched between each of the 36-inch-diameter flanges of the settling tank to reduce large scale turbulence. Each honeycomb is made of 1/4-inch cell

size and is 6 inches long in the flow direction. The resulting flow at the inlet pipe entrance was shown to have a velocity profile uniform within 1 percent across its diameter and a turbulence intensity of less than 0.5 percent, reference [2].

The NUWC transient flowmeter, presented in reference [3], is installed upstream of the inlet pipe and is an essential part of the facility since it provides instantaneous volumetric flow rate data. This meter was developed as part of the NUWC Unsteady Flow Loop Facility and has been shown to be accurate to ± 1 percent of reading in either steady-state or transient flows. There are no other known flowmeters with transient capability and high accuracy.

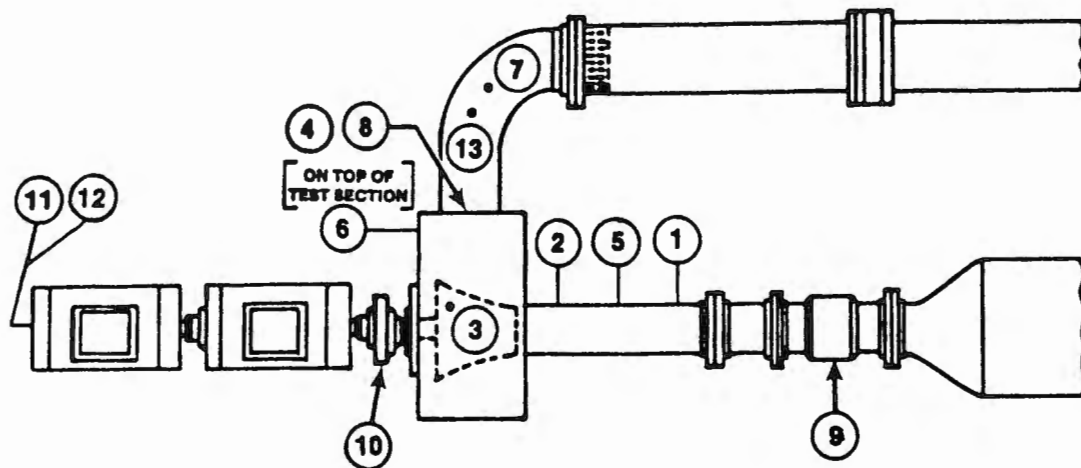
The facility can be pressurized to 110 psig for cavitation tests. A 2-inch diameter pipe is installed on the top of the test section to allow for pressurization of the facility through a hydraulic pump. The quality of the water is controlled by a filtering system which can filter particles down to 0.5 microns, a deaerator system to control the test fluid's (water) gas content, and a heat exchanger to control the temperature of the test fluid.

3. INSTRUMENTATION

The sensors and associated instrumentation installed in the facility provide the capability of assessing the hydrodynamic, cavitation, and hydro-acoustic performance of the impeller. The location of the various transducers are indicated in Figure 3-1.

For hydrodynamic performance, the sensors include four flush mounted pressure transducers, one temperature probe, a tachometer for impeller RPM, a transient electromagnetic flowmeter for instantaneous volumetric flowrate, and a torque meter installed between the motor and impeller to provide a direct measure of impeller shaft torque. Note in Figure 3-1 that pressure transducers are installed at two locations in the test section. One sensor is on top while the second one is in one of the side view ports. This allows for the identification of asymmetry in the discharge flow field, should any exist. Instrumentation for hydro-acoustic performance consists of three accelerometers. Table 3-1 lists the model numbers and operating ranges of each of the transducers, signal conditioners, and data acquisition hardware. Stated accuracy values are for the measurement and not the sensor alone since the values include propagation of errors from the signal conditioning, data acquisition, and the sensor.

Detailed acoustic evaluations are usually conducted on only the tape recorder data, which have the appropriate frequency bandwidth. The acoustic data can be recorded on the MASSCOMP to assure valid data had been acquired. The accelerometers are typically calibrated in accordance with standard procedures using a Bruel & Kjaer #4294 accelerometer calibrator outputting 1g acceleration at 159.2 Hz.



LOCATION	SENSOR
1 - 4	Flush Mounted Pressure Transducer
5 - 7	Piezoelectric Accelerometer
8	Thermister Probe
9	Transient Flowmeter
10	Torquemeter
11	Optical Rotary Encoder
12	Tachometer
13	Heise Gauge Pressure Transducer For Calibration

Figure 3-1. Instrumentation Locations

Table 3-1. Instrumentation

USE	HARDWARE	RANGE	ACCURACY	SIGNAL COND.
Pressure	Calibration Standard: Heise #623 Gauge	0.0 - 1.4 MPa	+/- 0.1 % of reading	Integral with Sensor
	Flush Mounted: Kulite XTM-190-100	0.0 - 2.8MPa	+/- 0.7 % of reading 0 - 80 kHz	Ectron 563 FL
Temperature	Cole-Palmer R8415-24 Thermister Probe	0.0 - 120.0 °C	+/- 0.05 °C	Cole-Palmer R8502-50
Transient Flow Rate	Foxboro Magnetic Flow Body with NUWC Electronics	0.0 - 40.0 fps	+/- 1.0 % of reading	N/A
Acceleration	Bruel & Kjaer 4371 or 4384 Accelerometer	0.1 - 12,600 Hz	1.0 +/- 2% $\mu\text{C/ms}^{-2}$	Ithaco #461 Charge Amplifier
Motor & Impeller Speed (RPM)	Radio-Energie Type RPY444RICB/YA Tachometer & Burton Mentor 55R Controller	0 - 2400 RPM	+/- 2 RPM	N/A
Impeller Position	Dynapar Model 63-C- ADF-0900-AO Optical Encoder	0 - 360 degrees	+/- 0.1 degree	C-Tek Model # LIN- 101-41-11-41-B Pulse Counter
Torque	HBM Type T2 Torquemeter	0 - 1500 lb-ft	+/- 1.5 lb-ft	N/A
Recorders	Honeywell 5600EC 14 Track Analog	N/A	5 - 20,000 Hz Wide Band Group I @ 30 ips	N/A
	1" Magnetic Tape Recorder	N/A	0 - 50 kHz @ 30 ips	N/A
	Masscomp 6600 Computer With Removable Hard Drive	N/A	1 Mhz Sample Rate to Memory 400 kHz to Hard Disk Total Sample Rate of 277k Samples/Sec	N/A
Monitors	Phillips Dual Channel Oscilloscope	N/A	DC - 20 Mhz	N/A

The transient flowmeter is comprised of three parts. The first is the flowmeter body which is installed in the piping of the facility and is commercially available. The second is the standard electronics provided with the meter body. These two components are used for steady-state measurement. The third component is the electronics developed by NUWC to provide the transient measurement capability.

The calibration of the flowmeter body had been conducted by the manufacturer. This provides the meter flow factor (GPM/Volts) which is the voltage output of the meter

body per unit flowrate. This value is in turn used to calibrate or set the span on the steady-state electronics provided with the meter. This is a standard procedure using the calibrator, Foxboro Model No. 8120 purchased with the flowmeter. This calibrated meter system is then ready for steady-state measurements and is used to calibrate the electronics which is used for transient measurements.

The electronics only requires that the desired upper range steady-state voltage output of the electronics (up to 10 volts) be adjusted to correspond to the maximum flowrate to be measured. This is accomplished by (1) operating the facility at its maximum flowrate, (2) measuring the maximum steady-state flowrate with the standard electronics, and (3) adjusting the electronics to the desired voltage. Since the meter and associated electronics are linear, the meter calibration factor, GPM/Volt, is calculated using this one data point. The actual linearity of the meter should be verified by comparing the output of the Foxboro and the output of the electronics at several flowrates spaced over the operating range.

The pressure transducers along with their signal conditioners are meant to be calibrated in-situ. The procedure is to install a highly accurate Heise Model No. 623 digital pressure gauge to the facility at the location of the pressure tap shown on Figure 3-1. This is then the standard to which the flush-mounted pressure transducers are compared at several different pressures over the operating range of the sensor (0 to 110 psig). Static pressure in the facility is adjustable by use of the hydraulic pump as described in the Facility Hardware and Control System Description section of this report.

The tachometer should be periodically calibrated at several motor rotational speeds by comparing the tachometer output to the actual speed as measured by a calibrated digital strobe tachometer. The torque meter was pre-calibrated at the factory.

4. FACILITY EVALUATION

As previously mentioned, the control system of the Impeller Test Facility has the capability to simultaneously control the impeller rotational speed, N , and the volumetric flowrate, Q . Thus the control systems allow for the precise modeling of any transient event desired. The input impeller rotational speed and flowrate versus time profiles used to model a typical transient impeller operation for torpedo ejection are shown in Figure 4-1. This figure shows both the user specified command signal from the input profiles and the resulting transient. The successful transient performance of the facility is manifested in the very close agreement between the command and actual curves for each of the control variables in the figure. Further proof of the ITF's performance is found in Figure 4-2, which shows the flow coefficient, C_q , of the transient versus time. Again there is very close agreement between the command signal and the actual transient data up to the time of simulated weapon exit, which is annotated on the figure.

An evaluation of the steady-state hydrodynamic performance of the facility can be determined from examination of Figure 4-3. In this figure, a plot of the head coefficient, C_h , versus C_q , is used to express the performance of a typical launch pump impeller installed in the facility. Data are presented for impeller rotational speeds of 500, 1000, 1500, 2000, and 2400 RPM. The least squares curve fit of the complete set of data at all rotational speeds is plotted as the solid line. The collapse of data onto a single curve for the different impeller speeds and the small amount of scatter provides confidence in the facility and associated instrumentation and confirms that there are negligible Reynolds number effects at all speeds tested and that the model scale data should provide a good representation of the full scale performance.

5. OPERATIONAL PROCEDURES

This section of the report presents the test procedures implemented for steady-state and transient operation of the Impeller Test Facility. The first subsection discusses general procedures that are common to both test modes. Procedures which are particular to only steady-state or transient operation are described in their respective sections. Furthermore, specific pre-test and test procedures are enumerated in their respective appendices at the end of the report.

GENERAL

The operational procedures for the Impeller Test Facility encompass the use of the facility hardware as well as the filtering and deaeration system. Prior to conducting any testing, the facility is filled with approximately 1500 gallons of tap water, which is passed through a 10-micron filter. After the facility has been filled, the water can be deaerated and recirculated through the 10-micron filter. Procedures for filtering and deaeration are listed in Appendix C, while a list of hardware checks and daily startup and shutdown procedures are listed in Appendix D.

STEADY-STATE TEST PROCEDURE

Steady-state testing may be performed to determine the following: establish impeller steady-state hydrodynamic performance characteristics, identify incipient cavitation, and to evaluate Reynolds number effects.

The steady-state hydrodynamic performance of the impeller can be determined by conducting tests for a wide-ranging set of RPM and flowrate combinations so that the complete flow coefficient range experienced during transient operation can be covered. For steady-state performance, the facility static pressure is set to a level high that eliminates the possibility of cavitation. Once the desired flowrate has been established through adjustment of the impeller speed and/or control valve, pressure, flowrate, torque,

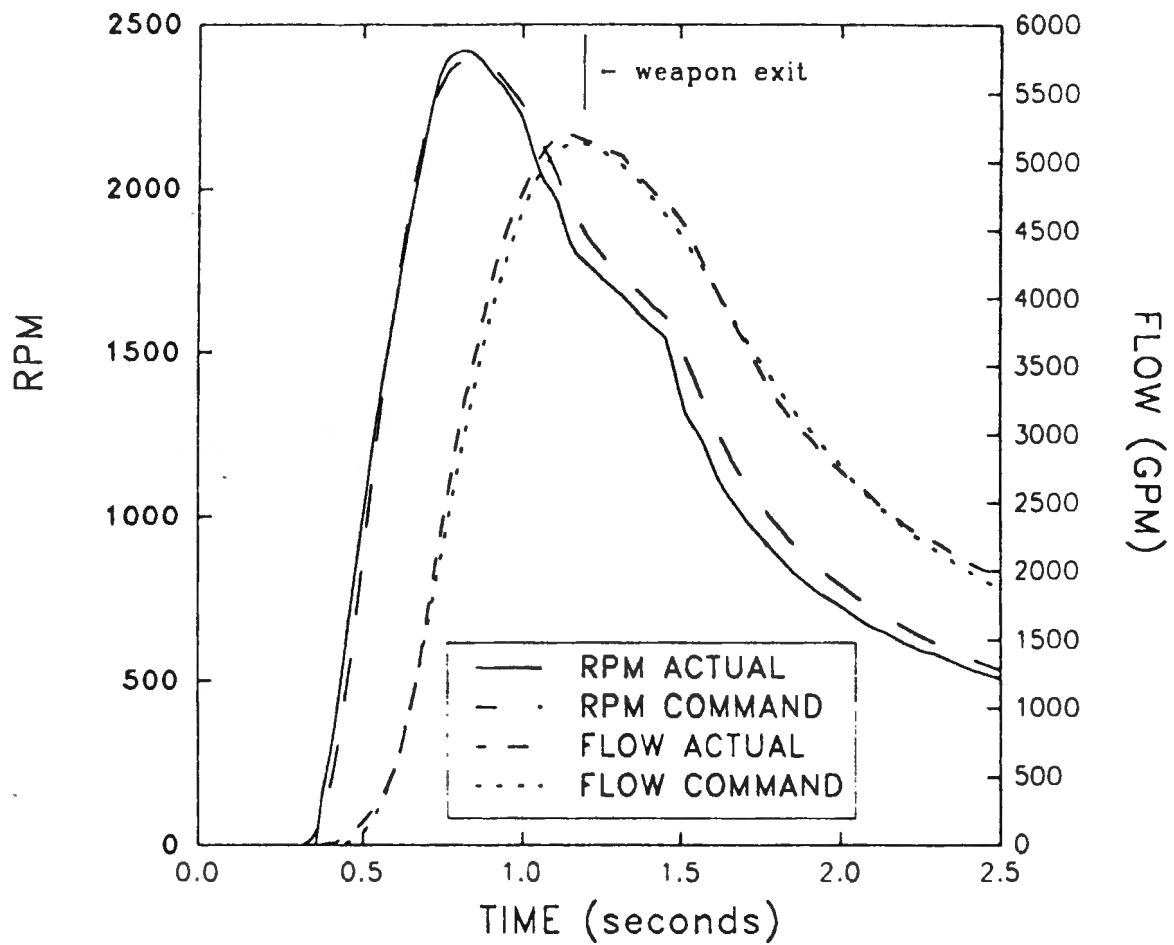


Figure 4-1. RPM & Flow Rate vs Time

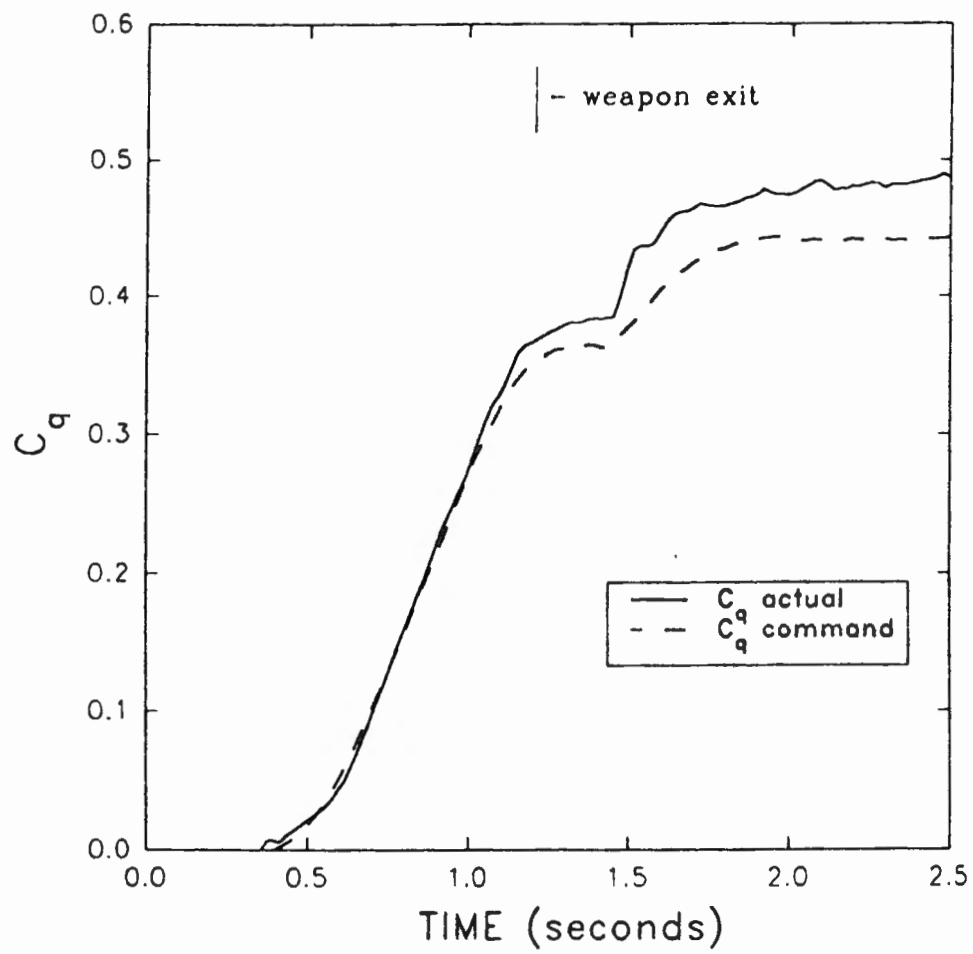


Figure 4-2. C_q vs Time

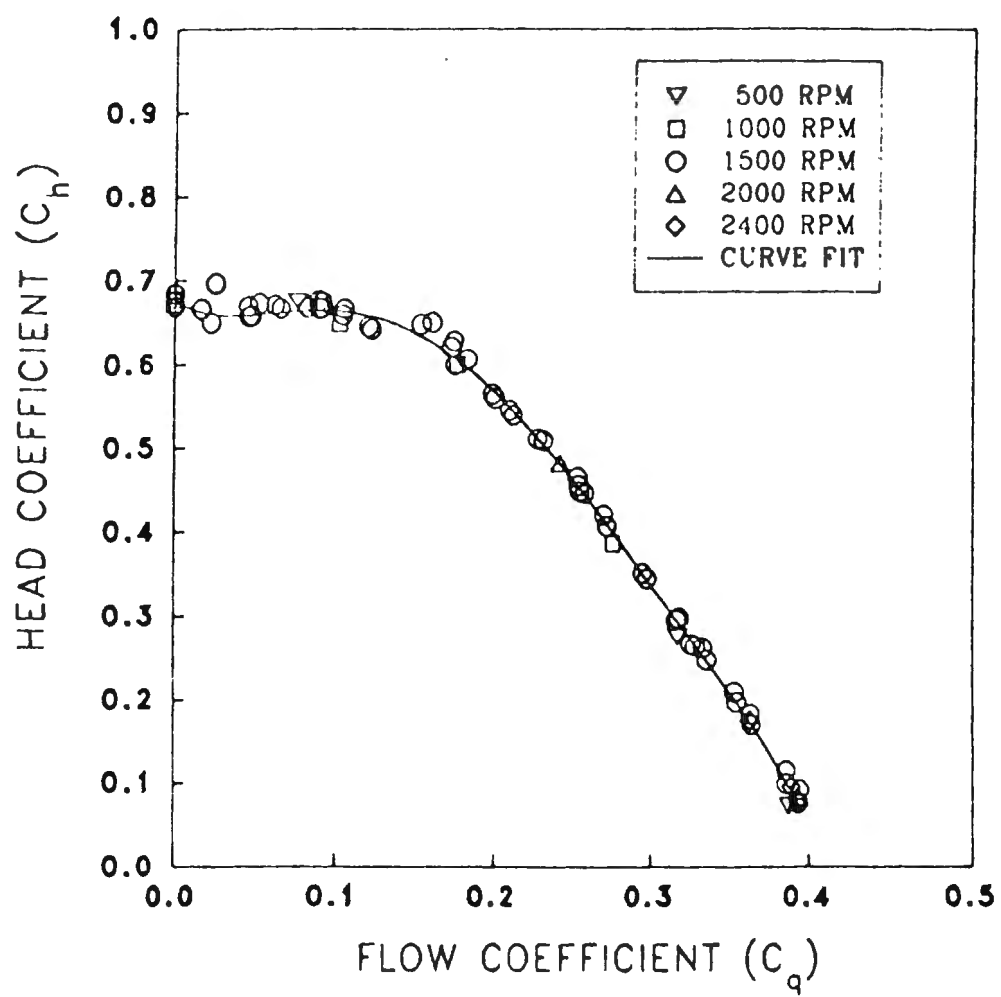


Figure 4-3. C_h vs C_q

and RPM data can be acquired by data acquisition software on the MASSCOMP 6600 computer described in the Data Acquisition section of this report.

Impeller incipient cavitation can be determined through either of two methods. The first method is to visually observe the formation of leading edge cavitation through the clear, acrylic inlet pipe of the facility. A strobe light or constant light source, coupled with an optical encoder, which serves to record the angular position of the impeller, and a camera can be used for this visualization technique. The second method is the observation of an increase in output level of the accelerometer mounted at station number 6 shown in Figure 3-1. This acoustic method is the only indicator of cavitation inception that can be used at higher C_q values where pressure side cavitation occurs and is not observable through the inlet pipe.

These cavitation experiments can be conducted for a wide range of flow coefficients. The facility static pressure is again set to such a level as to eliminate cavitation. Flowrate and RPM are then adjusted to give the desired flow coefficient. Facility pressure is then slowly lowered until cavitation occurs, at which point all relevant data such as upstream pressure, pump RPM, flowrate, and water temperature is recorded. See Appendix D for a step-by-step outline of steady-state operational procedures.

TRANSIENT TEST PROCEDURE

Transient test procedures require the use of the facility's motor and valve control systems, which operate simultaneously to produce a user specified impeller transient rotational speed (N) and flow rate (Q) time histories. Simultaneous control of both variables provides a means of properly modeling any submarine launch system by maintaining a one-to-one correspondence in flow coefficient, C_q , between the full-scale submarine and the Impeller Test Facility.

Both RPM versus time and valve position versus time data are downloaded to their respective computer. The facility is then pressurized to the desired static pressure. After facility pressurization, the control systems are activated as described in Section 2 of this report. When the control systems are triggered, the transient simulation is produced, while acoustic and/or performance data are acquired. The acoustic data are typically acquired on the 14-channel magnetic tape recorder described in the Instrumentation section of this report, and performance data are acquired on the MASSCOMP computer system. See Appendix D for a step-by-step outline of procedures required to conduct transient tests.

6. DATA ACQUISITION, REDUCTION, AND PRESENTATION

GENERAL

The acquisition of test data from the Impeller Test Facility is accomplished through the use of a MASSCOMP 6600 Data Acquisition System. The MASSCOMP system is activated by the trigger signal issued by the IBM AT during testing. The data acquisition is started 50 milliseconds before the motors are triggered and valve movement begins. Flowcharts of the data acquisition, reduction, and presentation process can be found in Appendix E.

The MASSCOMP is used to digitize and acquire data. The input range to the MASSCOMP is set to -5 to +5 volts. Data from each of the transducers except the temperature probe can be sampled at the frequency set by the data acquisition program ACQ.C, described below. Temperature is manually recorded at the beginning of a run. The Masscomp data can be immediately post-processed for a quick look capability to assure adequacy of the recorded data. At a later time, these data can be further reduced in detail for final analysis.

The accelerometer signals along with the common time reference signal from the IBM AT trigger are also recorded on magnetic tape for future data processing and archival storage. The recorder is a 14-track Honeywell 5600C analog magnetic tape recorder with 1-inch tape. Tape speed is set to 30 inches per second resulting in a frequency response of 20 kHz in Wide BAND-I. All channels are recorded in the FM mode. The output of each of the accelerometers' conditioning amplifier is set to provide a maximum of 1 Vrms output for the maximum expected response of the transducer.

Hydrodynamic data reduction is performed on the Launcher and Missile Systems Department's mv83a microvax 3400, using Fortran codes written especially for the Impeller Test Facility. The data presentation is performed with commercially available scientific software on a personal computer, as well as in-house developed graphics software used on the MASSCOMP 6600.

DATA ACQUISITION

The software used to acquire test data for the Impeller Test Facility is the NUWC developed program, ACQ.C, which is written in the C programming language. This code allows for a great deal of flexibility when setting up to acquire data. When the program is run, the user first enters the desired name of the test data file. Then the operator is given the choice of sampling data channels sequentially from a user defined starting channel number, or to randomly sample data if only a select number of channels is desired.

Next the operator is asked the number of analog and digital channels to be sampled, as well as channel numbers if random sampling is to be used. The sampling frequency is then input along with the duration of data acquisition in seconds and the time delay, in seconds, from where the trigger signal is received to the start of data collection. Following the input of all acquisition parameters, a screen display of all user inputs is given, including the number of all channels to be sampled. Here the operator may selectively change one or all of the previously selected parameters.

After the data acquisition parameters have been confirmed, the user is asked to type "GO" and to depress the carriage return key to place the MASSCOMP 6600 into a wait state. The data acquisition system remains in this wait state until it receives the low to high voltage signal sent by the IBM AT to initiate data acquisition.

When the program ACQ.C is executed, data are acquired sequentially and multiplexed. That is, for each sweep of the data channels during the acquisition (there is one sweep per sample period, i.e., a sampling rate of 1000 Hz means 1000 sweeps of the data channels per second) the data are sampled in order from the lowest channel number selected to the highest. This means that at the end of the data acquisition procedure, all data from all channels are lumped into one large output file. A program is then needed to demultiplex the data. That is, all the data from the first channel sampled must be written to its own data file and so on for all channels sampled.

The code that performs this task is DEMUX.C. This is another C language based program that was developed in-house specifically for use on the MASSCOMP. This code allows the demultiplexed data to be output in either of two forms. The first is engineering unit format where the data, for example, are in pounds per square inch for pressure data or gallons per minute for flowrate. The second output format is that used by Signal Technology's Interactive Laboratory System (ILS). This format is used for such signal processing techniques as Fast Fourier Transforms (FFT) of acoustic data files.

The user is asked to input the name of the raw data file given in the ACQ.C program. The code then asks in which of the aforementioned output formats the demultiplexed data are to be presented. Next the following data acquisition parameters are displayed on screen: number of channels sampled, duration of data acquisition, time delay used for data acquisition, and the sampling frequency. The channel numbers available for demultiplexing are given next along with the option of demultiplexing all channels displayed or a user defined number of channels, even a single channel if so desired. After the desired channels are selected, the calibration file, which provides for proper scaling of test data, is opened, and the data are demultiplexed and written to separate output files.

MASSCOMP DATA PRESENTATION

Once the original raw data file has been demultiplexed, the resulting data files are ready for presentation on the MASSCOMP 6600 graphics terminal or for output to a Hewlett Packard 7475 Plotter or IBM laser printer. The programs which perform these tasks are **PLOTOUT.C** and **PLOTENG.C**, which are initiated by the program **PAKMENU**. All of these codes, like those for data acquisition and demultiplexing, are written in C programming language.

The **PLOTOUT.C** code generates a set of three graphs which contain plots of data from full performance and acoustic testing. The name of the test run is input by the user, and the plot files are output with the test run name and the extensions **.graph1**, **.graph2**, and **.graph3**. Graph 1 displays the time traces of signals from accelerometers numbers 8, 9, and 11. Graph 2 shows the time traces of pressure data from pressure transducers numbers 1 through 6, and Graph 3 presents flowrate, tachometer, and torquemeter time traces. Graphs 1 through 3 may be displayed on the graphics terminal or a hardcopy may be made on the printer or plotter.

The **PAKMENU** program invokes the **PLOTENG.C** code, which is used to plot single engineering unit plots of data. This program is a very versatile tool for quick data output. The user is asked to enter the name of the test run for which data are to be plotted. The program then displays the following information about the test run: the data and time of data acquisition, the duration of the data acquisition in seconds, the time delay used in seconds, the number of channels sampled, and the sampling frequency.

The user is then asked for the time range the plot should encompass (beginning and ending times for the plot). After the time domain of the plot has been entered, the operator is presented with a listing of recorded variables available for plotting along with an option to return to the previous menu in order to change the run name or time range of the plot. This listing of recorded variables includes the demultiplexed channel numbers along with the instrumentation sampled on that channel. For example one choice might be the following: 4) Accelerometer #10. This choice would allow the user to plot the time trace of the signal output of Accelerometer #10, which was sampled on channel #4.

In the event the user selects a data file which has not been demultiplexed, this program exits to the **DEMUX.C** program, demultiplexes the raw data file, and then returns to the **PLOTENG.C** program. Then the operator is asked to input the number of initial data points to be averaged to rezero the test data. If zero is entered, the data are not rezeroed. Acoustic data are not usually rezeroed, whereas performance is customarily rezeroed for graphical output.

Next the program presents a menu that allows the user to average or increment the test data, if so desired. If the option to average data is selected, the user is requested to enter the number of data points to be averaged together. For example, if the number of points to be averaged is 50, then every 50 data points throughout the data file would be averaged together and placed at a single point in the time trace. This option is very

useful when attempting to smooth out data affected by signal noise during acquisition. If the option to increment data is selected, the number of points to skip is input. If the increment entered is 50, the program will plot every 50th point in the data file instead of averaging 50 points.

Finally the plot is output to the graphics terminal for viewing. Besides the time trace of the selected data the plot also displays the appropriated axes titles, the date and time of data acquisition, the date and time of plot generation, and the name of the instrumentation from which data were collected (i.e., Accelerometer #8, etc.). The user is then asked if a hardcopy is required. The **PLOTENG.C** code is often used as a quick and efficient method of checking the gain settings for the accelerometers of the ITF.

DATA REDUCTION

Before test data can be reduced to give such impeller performance results such as the coefficient of flow, C_q , or the head coefficient, C_h , the data must be converted from binary to ASCII format. This is performed on the MASSCOMP 6600, using the batch program **TRBTOA**, which uses the built in **btoa** command to perform the conversion on all performance data required. Once the conversion is completed the data files are uploaded to the Launcher and Missile System Department's Digital Equipment Corporation's mv83a microvax 3400 through the use of FTP (File Transfer Program). The **mget** command in FTP allows the user to transfer all of the converted data files with the same file extension to the microvax, while the **get** command transfers files one at a time. These data files are then processed using a variety of programs written in the Fortran program language. These programs are as follows: **FLOW1**, **FLOW2**, **TRZERO**, **TRPERF**, **TRPERF2**, **DELPP1P2**, and **DELP3P6**.

The **FLOW1.FOR** program is used to read a data file from a transient test run, calculate the time for each data point in the file, and outputs the point number, time, and flowrate to a new data file. The user is asked to input the name of the file to be processed, the desired name of the output file, and the sampling rate used to acquire the data in the file. After this has been run, the output file is analyzed to determine at what point number the transient begins. When this point has been determined, the data are ready for the next step in the process.

The next step in the data reduction process is to run the program **FLOW2.FOR**. The function of this code is to interpolate the raw data obtained from the transient flowmeter during a transient run of the Impeller Test Facility. The flowmeter is designed to update at a rate of 60 Hz. Since the data sampling rate during testing is usually much higher than the update rate of the flowmeter, groups of data with approximately the same value (the slight differences observed are due to data noise) appear between update points. To correct for the lower update rate of the flowmeter, this program takes each set of data points between updates, determines the average value of these points, and assigns the average value to the lower update point. Once update points have been corrected for

noise, the program interpolates between the update points. The output of **FLOW2.FOR** is a file with the same format as the input file. For this program the user is asked to enter the input filename, output filename, the data sampling rate, and the data point number at the start of the transient as determined from the use of **FLOW1.FOR**.

Now the program **TRZERO.FOR** is used. This program is designed to calculate and apply the zeros for impeller performance data taken during a transient test run, if the data has any DC shift. The user is asked to input the run number, run name, and the number of points to be used for the zero average.

Once the acquired test data have been passed through the three previously mentioned programs, **TRPERF2.FOR** is run. This program is used as a means of calculating various impeller performance data from a transient test run, utilizing the following test data: flowrate, tachometer data (RPM), torquemeter data, and data from pressure transducers numbers 2 and 6. This program is used with data that are clean relative to the noise level found in the data (i.e., the data has an acceptable signal-to-noise ratio). If the data are noisy, **TRPERF.FOR**, which will be discussed below, must be used.

The user is first asked to input the run number. This number will be used as the file extension of all files output by **TRPERF2.FOR**. Next the run name prefix is entered (i.e., tr, imp, etc.) along with the data sampling rate. The user then inputs the increment step (i.e., the number of points to skip), the number of impeller on which the run was conducted, and the vapor pressure of water, in psia, at the test temperature.

Once all user inputs are entered, the program first creates arrays of data incremented by the input increment step. These data are then used to calculate the performance characteristics of the impeller. The following impeller performance characteristics are then calculated and written to their respective output files: head; flowrate; C_h , the head coefficient; C_q , the flow coefficient; torque; transient and quasi-steady-state cavitation numbers; pump RPM; efficiency; and acceleration. The output files from this program can be used for presentation of data upon transferring the data to a PC using the **@userlib:vaxtopc** command on the microvax. This command allows the user to transfer the performance data files singly or by using the wildcard notation (*.for, *.123, cavss.*, etc.).

The program **TRPERF.FOR** is used to calculate the same impeller performance data throughout a transient launch as **TRPERF2.FOR**, but it is used on data that contain excessive noise and requires averaging. This code asks for all of the same user inputs as **TRPERF2.FOR** along with a few other inputs. These additional inputs are the number of points to average in the flowmeter data file, and the number of points to average for all other data, if averaging is indeed desired. The output impeller performance characteristics are the same as those output by **TRPERF2.FOR**.

The two computer codes **DELP1P2.FOR** and **DELP3P6.FOR** are nearly identical. The only difference is that **DELP1P2.FOR** is used to determine the pressure difference between the two inlet pipe pressure transducers (P1 and P2), while **DELP3P6.FOR** is utilized to determine the pressure difference between the two test section pressure transducers (P3 and P6) during transient test runs. Both programs ask the user for the run number and name, the data sampling rate, and the number of points to average or increment, if so desired.

PERFORMANCE DATA PRESENTATION

Once the files containing the impeller performance characteristics calculated by either **TRPERF.FOR** or **TRPERF2.FOR** are transferred to a PC using the **@userlib:vaxtopc** command, these data files may be used for graphical presentation using Jandel Scientific Sigma Plot.

Sigma Plot is a very flexible graphical presentation program that allows the user to display pump impeller performance data in the most edifying format possible. The flexibility of this package allows the generation of plots with multiple x and y axes, enabling a single graph to display several plots at once. For example plots of head, flowrate, RPM, and C_h can all be displayed on a single graph, or simple plots such as steady-state and transient cavitation numbers versus time, or C_q versus C_h .

7. REFERENCES

1. P. J. Lefebvre, and K. M. LaPointe, "Flow Loop Facility Flow Control System," NUSC Technical Memorandum 85-2076, Naval Underwater Systems Center, Newport, RI, 1985 (UNCLASSIFIED).
2. P. J. Lefebvre, "Design and Evaluation of NUSC's Flow Loop Facility," NUSC Technical Document 86-6512, Naval Underwater Systems Center, Newport, RI, 1986 (UNCLASSIFIED).
3. P. J. Lefebvre, and W. W. Durgin, W. W., "A Transient Electromagnetic Flowmeter and Calibration Facility," *Journal of Fluids Engineering*, March 1990.

APPENDIX A: CONTROL SYSTEMS SOURCE CODES

MOTORS.BAS

VCONIMPL.BAS

The following program is MOTORS.BAS which controls the DC motors:

```

*****
'*          N.U.S.C. - JOB # 669238A          *
'*          DECEMBER 12, 1989 - RAP          *
'*          MENTOR COMMUNICATIONS PROGRAM      *
'*          FOR THE IBM PC/XT/AT OPERATING UNDER MS-DOS *
'*          FOR USE WITH PROC1 V1.8G AND ANSI COMMS *
'*                                           *
'*  THERE ARE VARIOUS .DAT DATA FILES REQUIRED FOR THIS *
'*  PROGRAM TO RUN AND WILL BE CREATED USING DEFAULTS *
'*  IF NOT FOUND. *
*****
'
*****
'*  DEFINE AND INITIALIZE ALL VARIABLES AND DEFAULTS *
*****

ON ERROR GOTO ErrorTrap
SCREEN 0,0: WIDTH 80
KEY OFF: CLOSE
DEFINT A-Z
DIM PARAM22$(200)    '** STORAGE FOR STRING VALUES
DIM PARAM22!(200)    '** STORAGE FOR REAL VALUES LOADED FROM
DISK
DIM CS22%(200)        '** STORAGE FOR CALCULATED CHECKSUMS
RESETVAL$ = "+255"    '** RESET VALUE FOR PARAMETER #099
RESETPARM$ = "099"    '** RESET PARAMETER #
THYRISTOFF$ = "+001"  '** THYRISTOR TURN OFF VALUE
THYRISTON$ = "+000"   '** THYRISTOR TURN ON VALUE
THYRISTPARM$ = "166"  '** THYRISTOR ON/OFF PARAMETER #
INTREFVAL$ = "+001"   '** INTERNAL REF. VALUE
EXTREFVAL$ = "+000"   '** EXTERNAL REF. VALUE
EXTINTPARM$ = "155"   '** SELECT EXTERNAL/INTERNAL
REF.PARAMETER #
SPEEDPARM$ = "022"    '** SPEED REF. PARAMETER #
SPEEDREFZERO$ = "+000" '** INTERNAL SPEED REF. VAL OF 0
RET$ = CHR$(17) + CHR$(196) + CHR$(217)    '** GRAPHIC RETURN KEY
DFLAG = 0          '**** FLAG USED FOR .DAT FILE DETECTION OPEN ON
ERROR
SFLAG = 0          '**** FLAG USED FOR SHELNAME.DAT FILE DETECTION
'
'***** INITIALIZE DRIVE NUMBER *****
'
AD = 1 'DRIVE NUMBER DEFAULT

```

```

ADMSB$ = "0"
ADLSB$ = "1"
AD$ = ADMSB$ + ADLSB$
,

'***** GET ALL .DAT FILES FROM DISK ***
,

OPEN "ABAUD.DAT" FOR INPUT AS #1: INPUT #1, BAUD$: CLOSE #1
SELECT CASE BAUD$
    CASE "300": BAUDBOT$ = " 300"    '* BAUDBOT$ USED IN
    CASE "600": BAUDBOT$ = " 600"    '* BOTTOM LINE 25
    CASE "1200": BAUDBOT$ = " 1200"
    CASE "2400": BAUDBOT$ = " 2400"
    CASE "4800": BAUDBOT$ = " 4800"
    CASE "9600": BAUDBOT$ = " 9600"
    CASE "19200": BAUDBOT$ = "19200"
END SELECT
OPEN "ACOM.DAT" FOR INPUT AS #1: INPUT #1, MCOM$: CLOSE #1
OPEN "RESP.DAT" FOR INPUT AS #1: INPUT #1, RESPONSE&: CLOSE #1
OPEN "CRESP.DAT" FOR INPUT AS #1: INPUT #1, CRESPONSE&: CLOSE #1
OPEN "ZRESP.DAT" FOR INPUT AS #1: INPUT #1, ZERORESP&: CLOSE #1
,

DFLAG = 1 '***** SET .DAT FLAG - ALL FILE OPEN's PASSED THIS POINT **
,

'*** CONFIGURE COM(n) PORT FOR MENTOR & THEN RUN MAIN MENU
***
MainMenu:
    PARITYDATASTOP$ = "E,7,1"
    COMFIL$ = "COM" + MCOM$ + ":" + BAUD$ + "," + PARITYDATASTOP$ +
    ",CS,DS"
    CLOSE
,

'*****
'*          DISPLAY MAIN MENU          *
'*****

FILLER$ = "    * MAIN MENU *    "
GOSUB BottomLine25: COLOR 15, 0: CLS 2
LOCATE 4, 22, 1: PRINT " NAVAL UNDERWATER SYSTEMS CENTER  "
LOCATE 5, 22: PRINT " CONTROL TECHNIQUES JOB No. 669238A  "
LOCATE 6, 22: PRINT " MENTOR COMMS/ANSI COMMS PROTOCOL  "
LOCATE 7, 22: PRINT STRING$(36, 95)
LOCATE 9, 22: PRINT "P. PARAMETER ACCESS, READ/WRITE TO DRIVE"
LOCATE 10, 22: PRINT "U. UPLOAD THE PROFILE TO THE DRIVE"
LOCATE 11, 22: PRINT "L. LOAD A PROFILE INTO MEMORY FROM DISK"
LOCATE 13, 22: PRINT "R. RESET THE DRIVE"
LOCATE 14, 22: PRINT "T. TEST A DUMMY PROFILE TO DRIVE"

```

```

LOCATE 15, 22: PRINT "Z. ADJUST ZERO SPEED LOOP TIMEOUT VALUE"
LOCATE 16, 22: PRINT "O. ADJUST ON COM(n) PORT TIMEOUT VALUE"
LOCATE 17, 22: PRINT "A. ADJUST TRANSMIT LOOP RESPONSE TIME
VALUE"

```

```

LOCATE 18, 22: PRINT "C. CONFIGURE BAUD & COMPORT"

```

```

LOCATE 19, 22: PRINT "S. SHELL TO A PROGRAM"

```

```

LOCATE 20, 22: PRINT "Q. QUIT TO DOS"

```

```

LOCATE 22, 23, 1, 0, 7: INPUT ; " Enter your selection: ", N$

```

```

SELECT CASE UCASE$(N$)

```

```

    CASE "P": GOTO ParameterAccess

```

```

    CASE "U": GOTO Upload

```

```

    CASE "L": GOTO LoadProfile

```

```

    CASE "R": GOTO ResetDrive

```

```

    CASE "T": GOTO TestLoad

```

```

    CASE "Z": GOTO ZERORESP

```

```

    CASE "O": GOTO OnComResp

```

```

    CASE "A": GOTO Adjustresp

```

```

    CASE "C": GOTO ChangeBaudCom

```

```

    CASE "S": GOTO ShellToProg

```

```

    CASE "Q": GOTO QuitToDOS

```

```

    CASE ELSE: GOTO MainMenu

```

```

END SELECT

```

```

GOTO MainMenu

```

```

*****
'*                                     *
'*      P.      PARAMETER ACCESS (SUB)      *
'*                                     *
'*                                     *
*****

```

```

ParameterAccess:

```

```

    FILLER$ = " PARAMETER ACCESS, R/W "

```

```

    GOSUB BottomLine25: COLOR 7, 0

```

```

ParameterAccess1:

```

```

    GOSUB TurnOnCom

```

```

    SCROLLING = 0

```

```

    CLS 2

```

```

    LOCATE 6, 20: PRINT "R = Read a Parameter"

```

```

    LOCATE 8, 20: PRINT "W = Write a Parameter"

```

```

    LOCATE 10, 20: PRINT "S = Scroll through Parameters"

```

```

    LOCATE 12, 20: PRINT "ESC = Return to main menu"

```

```

    LOCATE 14, 1: PRINT STRING$(75, 32)

```

```

    LOCATE 14, 20, 1

```

```

    PRINT "Enter required option: ";

```

GetRWS:

```
A$ = INKEY$: IF A$ = "" THEN GOTO GetRWS
LOCATE 20, 1: PRINT STRING$(70, 32)
LOCATE 22, 1: PRINT STRING$(70, 32)
LOCATE 14, 43: PRINT A$;
SELECT CASE UCASE$(A$)
    CASE CHR$(27): GOTO MainMenu
    CASE "R": GOTO ReadParameter
    CASE "W": GOTO WriteParameter
    CASE "S": GOTO ScrollParameter
    CASE ELSE: LOCATE 14, 43: GOTO GetRWS
END SELECT
```

ScrollParameter:

```
LOCATE 20, 20
INPUT "Which Parameter do you wish to start at: ", PP$
IF PP$ = "" THEN GOTO ParameterAccess1
SELECT CASE VAL(PP$)
    CASE 1 TO 255: P = VAL(PP$)
    CASE ELSE
        LOCATE 20, 1
        PRINT STRING$(75, 32)
        GOTO ScrollParameter
END SELECT
GOSUB PADP
CLS 2
GOSUB ReadPacket
LOCATE 22, 1
PRINT TAB(5); "#" + P$; " has a value of: "; V$
LOCATE 10, 1
PRINT TAB(20); "Space    = Read same Parameter again"
PRINT
PRINT TAB(20); "Return    = Read next Parameter"
PRINT
PRINT TAB(20); "Back-Space = Read previous Parameter"
PRINT
PRINT TAB(20); "R        = Return to previous menu"
```

Scroll1:

```
SCROLLING = 0
LOCATE 20, 1, 1: PRINT TAB(20); "Enter required option ? ";
```

Scroll2:

```
O$ = INKEY$: IF O$ = "" THEN GOTO Scroll2
IF ASC(O$) <> 13 AND UCASE$(O$) <> "R" THEN
    IF ASC(O$) <> 8 AND ASC(O$) <> 32 THEN GOTO Scroll1
END IF
IF UCASE$(O$) = "R" THEN GOTO ParameterAccess1
SCROLLING = 1
```

```

GOSUB ReadPacket
LOCATE 22, 1
PRINT STRING$(79, 32)
LOCATE 22, 1
PRINT TAB(5); "#" + P$; " has a value of: "; V$
GOTO Scroll1
WriteParameter:
LOCATE 20, 20
INPUT "Enter PARAMETER you wish to write: ", PP$
IF PP$ = "" THEN GOTO ParameterAccess1
SELECT CASE VAL(PP$)
CASE 1 TO 255: P = VAL(PP$)
CASE ELSE
LOCATE 20, 1
PRINT STRING$(75, 32)
GOTO WriteParameter
END SELECT
GOSUB PADP
GOSUB ReadPacket
LOCATE 22, 1
PRINT STRING$(79, 32)
LOCATE 22, 1
PRINT TAB(5); "#"; P$; " has a value of: "; V$
IF GOODREAD = 1 THEN GOTO WriteValue
GOTO WriteParameter
WriteValue:
LOCATE 20, 1
PRINT STRING$(75, 32)
LOCATE 20, 20
INPUT "Enter VALUE you wish to write: ", W$
IF W$ = "" THEN GOTO ParameterAccess1
V = VAL(W$)
IF V > 999 OR V < -999 THEN GOTO WriteValue
GOSUB PADV
GOSUB CalcChksumWrite
LOCATE 22, 1
PRINT STRING$(79, 32)
LOCATE 22, 1
PRINT TAB(5); "#" + P$; TAB(11); V$; " ";
GOSUB WritePacket
LOCATE 20, 1
PRINT STRING$(75, 32)
GOTO WriteParameter
ReadParameter:
LOCATE 20, 20
INPUT "Enter Parameter you wish to read: ", PP$

```

```

IF PP$ = "" THEN GOTO ParameterAccess1
SELECT CASE VAL(PP$)
CASE 1 TO 255: P = VAL(PP$)
CASE ELSE
LOCATE 20, 1
PRINT STRING$(75, 32)
GOTO ReadParameter
END SELECT
GOSUB PADP
GOSUB ReadPacket
LOCATE 22, 1
PRINT STRING$(79, 32)
LOCATE 22, 1
PRINT TAB(5); "#" + P$; " has a value of: "; V$
LOCATE 20, 1
PRINT STRING$(75, 32)
GOTO ReadParameter
,
*****
'*   L.   LOAD A PROFILE TO MEMORY FROM DISK   *
*****
,
LoadProfile:
FILLER$ = " LOAD PROFILE TO MEMORY"
GOSUB BottomLine25: COLOR 7, 0: CLS 2
FLOAD = 0
LOADFILNUM = FREEFILE
LOCATE 15, 25: LINE INPUT ; "ENTER PROFILE FILENAME.EXT: ";
PROFILE$
OPEN PROFILE$ FOR INPUT AS #LOADFILNUM
CLS 2
I = 0
WHILE NOT EOF(LOADFILNUM)
I = I + 1
INPUT #LOADFILNUM, PARAM22!(I)
PRINT PARAM22!(I)
WEND
PARAM22!(I + 1) = -1
CLOSE #LOADFILNUM
LOCATE 15, 25
PRINT USING "LOADED ### VALUES FROM FILE: &"; I; PROFILE$
LOCATE 20, 25
PRINT "PRESS ANY KEY TO BEGIN CONVERTING VALUES TO STRINGS"
Convertval:
A$ = INKEY$: IF A$ = "" THEN GOTO Convertval
CLS 2

```

```

LOCATE 24, 1
P$ = SPEEDPARM$ '** PARAMETER #22
I = 1
WHILE PARAM22!(I) <> -1
  V = INT(PARAM22!(I) / 2400 * 999)
  IF V > 999 THEN V = 999
  GOSUB PADV
  PARAM22$(I) = V$
  GOSUB CalcChksumWrite
  CS22%(I) = CS
  PRINT "REAL = "; PARAM22!(I); TAB(18); " INT = "; V; TAB(33);
  PRINT " STRING VALUE = "; PARAM22$(I); TAB(58); " CHKSUM = ";
  CS22%(I)
  SELECT CASE I
    CASE 20, 40, 60, 80:
      PRINT : PRINT " "; I; " DONE.. ANY KEY TO CONTINUE....."
      GOTO Pause20
    CASE ELSE: GOTO Skip20
  END SELECT
Pause20:
  A$ = INKEY$: IF A$ = "" GOTO Pause20
  PRINT
Skip20:
  I = I + 1
WEND
  V = PARAM22!(I)
  GOSUB PADV
  PARAM22$(I) = V$
  GOSUB CalcChksumWrite
  CS22%(I) = CS
  PRINT "REAL = "; PARAM22!(I); TAB(18); " INT = "; V; TAB(33);
  PRINT " STRING VALUE = "; PARAM22$(I); TAB(58); " CHKSUM = ";
  CS22%(I)
  PRINT : PRINT " "; I - 1; " DONE.. END.... ANY KEY TO CONTINUE....."
PauseLast:
  A$ = INKEY$: IF A$ = "" GOTO PauseLast
  FLOAD = 1
  GOTO MainMenu
,
,

```

```

'*****
'*      U.   UPLOAD A PROFILE TO THE DRIVE      *
'*****
,

```

```

Upload:
  FILLER$ = "  UPLOAD A PROFILE  "

```

```

GOSUB BottomLine25: COLOR 7, 0: CLS 2
IF FLOAD = 0 THEN
    LOCATE 13, 20: PRINT "NO PROFILE LOADED....."
    LOCATE 15, 20: PRINT "PRESS 'ESC' FOR MainMenu"
UpPause1:
    A$ = INKEY$: IF A$ = "" THEN GOTO UpPause1
    IF A$ <> CHR$(27) THEN GOTO UpPause1 ELSE GOTO MainMenu
END IF
GOSUB TurnOnCom
IF MCOM$ = "2" THEN ACOM$ = "1" ELSE ACOM$ = "2"
P$ = SPEEDPARM$: V$ = SPEEDREFZERO$: GOSUB CalcChksumWrite
LOCATE 18, 1
PRINT TAB(10); "#"; P$; TAB(16); V$; "  ";
GOSUB WritePacket
PRINT
PRINT TAB(10); "PRESS ANY KEY WHEN READY TO START....."
PRINT TAB(10); "OR PRESS 'ESC' FOR MainMenu"
PRINT
UpPause2:
    A$ = INKEY$: IF A$ = "" THEN GOTO UpPause2
    IF A$ = CHR$(27) THEN GOTO MainMenu
    P$ = EXTINTPARM$: V$ = INTREFVAL$: GOSUB CalcChksumWrite
    PRINT TAB(10); "#"; P$; TAB(16); V$; "  ";
    GOSUB WritePacket
    PRINT
    PRINT TAB(10); "WAITING FOR TRIGGER FROM AT... (60 Sec Limit).."
    PRINT
    COM(VAL(MCOM$)) OFF
    P$ = SPEEDPARM$
    APPLEFILNUM = FREEFILE
Trigger:
    OPEN "COM" + ACOM$ + ":9600,N,7,1" FOR INPUT AS #APPLEFILNUM
    LET A$ = "N"
    DO WHILE (A$ <> "A")
        A$ = INPUT$(1, #APPLEFILNUM)
    LOOP
    CLOSE #APPLEFILNUM
    PRINT "<-----TRIGGER RECEIVED-----"
>"
    I = 1
    GETTIME1! = TIMER
    WHILE VAL(PARAM22$(I)) <> -1
        PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
        PRINT #1, P$; PARAM22$(I); CHR$(3); CHR$(CS22%(I));
        GOSUB TimeLoop
        PRINT ".";

```



```

    I = I + 1
WEND
GETTIME2! = TIMER
P$ = THYRISTPARM$: V$ = THYRISTOFF$: GOSUB CalcChksumWrite
PRINT #1, CHR$(4); ADMSB$: ADMSB$: ADLSB$: ADLSB$: CHR$(2);
PRINT #1, P$: V$: CHR$(3); CHR$(CS);
PRINT P$; "="; V$
PRINT "POINTS = "; I - 1; " UPLOAD TIME IN SECONDS/POINT = ";
PRINT ((GETTIME2! - GETTIME1!) / (I - 1))
GOSUB TurnOnCom
PRINT
PRINT TAB(10); "#"; P$: TAB(16); V$: " ";
GOSUB WritePacket
P$ = SPEEDPARM$: V$ = SPEEDREFZERO$: GOSUB CalcChksumWrite
PRINT TAB(10); "#"; P$: TAB(16); V$: " ";
GOSUB WritePacket
P$ = EXTINTPARM$: V$ = EXTREFVAL$: GOSUB CalcChksumWrite
PRINT TAB(10); "#"; P$: TAB(16); V$: " ";
GOSUB WritePacket
PRINT
PRINT TAB(10);
PRINT "DOWNLOAD COMPLETE... WAITING FOR ZERO SPEED LOOP
TIMEOUT..."
FOR W& = 1 TO ZERORESP&
LOCATE 23, 70, 0
PRINT "\";
LOCATE 23, 70, 0
PRINT "I";
LOCATE 23, 70, 0
PRINT "/";
NEXT W&
LOCATE 23, 70, 0
PRINT " "
PRINT
PRINT TAB(10); "TIMOUT COMPLETE... ENABLING THYRISTOR
BRIDGE....."
PRINT
P$ = THYRISTPARM$: V$ = THYRISTON$: GOSUB CalcChksumWrite
PRINT TAB(10); "#"; P$: TAB(16); V$: " ";
GOSUB WritePacket
PRINT
PRINT TAB(10); "ANY KEY TO RETURN TO MAIN MENU....."
UpPause3:
A$ = INKEY$: IF A$ = "" THEN GOTO UpPause3
GOTO MainMenu

```

```

'*****
'*   R.   RESET THE DRIVE                               *
'*****

```

ResetDrive:

```

FILLER$ = " RESET THE DRIVE  "
GOSUB BottomLine25: COLOR 7, 0: CLS 2
LOCATE 12, 10
PRINT "ARE YOU SURE THE DRIVE IS AT ZERO (0) SPEED...(Y/N).?? ";

```

ResetWait1:

```

A$ = INKEY$: IF A$ = "" THEN GOTO ResetWait1
IF UCASE$(A$) = "Y" THEN
    GOSUB TurnOnCom
    PRINT
    PRINT
    P$ = RESETPARM$: V$ = RESETVAL$: GOSUB CalcChksumWrite
    PRINT TAB(10); "#"; P$; TAB(16); V$; "  ";
    GOSUB WritePacket
    PRINT
    PRINT TAB(10); "ANY KEY FOR MAIN MENU..."

```

ResetWait2:

```

A$ = INKEY$: IF A$ = "" THEN GOTO ResetWait2
END IF
GOTO MainMenu

```

```

'*****
'*   T.   TEST A DUMMY PROFILE TO DRIVE                     *
'*****

```

TestLoad:

```

FILLER$ = " TEST DUMMY PROFILE  "
GOSUB BottomLine25: COLOR 7, 0: CLS 2
GOSUB TurnOnCom
P$ = SPEEDPARM$: V$ = SPEEDREFZERO$: GOSUB CalcChksumWrite
LOCATE 18, 1
PRINT TAB(10); "#"; P$; TAB(16); V$; "  ";
GOSUB

```

WritePacket

```

PRINT
PRINT TAB(10); "PRESS, (Z) FOR ZERO RAMP, (R) FOR RAMP 0-999-0"
PRINT TAB(10); "OR PRESS 'ESC' FOR MainMenu"
PRINT

```

TestPause2:

```

A$ = INKEY$: IF A$ = "" THEN GOTO TestPause2

```

```

SELECT CASE UCASE$(A$)
  CASE CHR$(27): GOTO MainMenu
  CASE "Z": GOTO TestZero
  CASE "R": GOTO TestRamp
  CASE ELSE: GOTO TestPause2
END SELECT
TestZero:
  PRINT
  P$ = EXTINTPARM$: V$ = INTREFVAL$: GOSUB CalcChksumWrite
  PRINT TAB(10); "#"; P$; TAB(16); V$; " ";
  GOSUB WritePacket
  PRINT
  D = 1
  P$ = SPEEDPARM$: V$ = SPEEDREFZERO$: CS = 40
  COM(VAL(MCOM$)) OFF
  WHILE D <> 999
    PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
    PRINT #1, P$; V$; CHR$(3); CHR$(CS);
    GOSUB TimeLoop
    PRINT ".";
    D = D + 1
  WEND
  WHILE D <> 0
    PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
    PRINT #1, P$; V$; CHR$(3); CHR$(CS);
    GOSUB TimeLoop
    PRINT ".";
    D = D - 1
  WEND
  GOTO TestEnd
TestRamp:
  PRINT
  P$ = EXTINTPARM$: V$ = INTREFVAL$: GOSUB CalcChksumWrite
  PRINT TAB(10); "#"; P$; TAB(16); V$; " ";
  GOSUB WritePacket
  PRINT
  D = 1
  COM(VAL(MCOM$)) OFF
  WHILE D <> 999
    P$ = SPEEDPARM$: V = D: GOSUB PADV: GOSUB CalcChksumWrite
    PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
    PRINT #1, P$; V$; CHR$(3); CHR$(CS);
    GOSUB TimeLoop
    PRINT ".";
    D = D + 1
  WEND

```

```

WHILE D <> 0
    P$ = SPEEDPARM$: V = D: GOSUB PADV: GOSUB CalcChksumWrite
    PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
    PRINT #1, P$; V$; CHR$(3); CHR$(CS);
    GOSUB TimeLoop
    PRINT ". ";
    D = D - 1
WEND
TestEnd:
    GOSUB TurnOnCom
    PRINT : PRINT
    P$ = SPEEDPARM$: V$ = SPEEDREFZERO$: GOSUB CalcChksumWrite
    PRINT TAB(10); "#"; P$; TAB(16); V$; " ";
    GOSUB WritePacket
    P$ = EXTINTPARM$: V$ = EXTREFVAL$: GOSUB CalcChksumWrite
    PRINT TAB(10); "#"; P$; TAB(16); V$; " ";
    GOSUB WritePacket
    PRINT : PRINT
    PRINT TAB(10); "PRESS ANY KEY FOR Main Menu...."
TestPause3:
    A$ = INKEY$: IF A$ = "" THEN GOTO TestPause3
    GOTO MainMenu
'
'*****
'*   A.   ADJUST TRANSMIT LOOP RESPONSE TIME VALUE   *
'*****
Adjustresp:
    FILLER$ = " XMIT LOOP RESPONSE  "
    GOSUB BottomLine25: COLOR 7, 0: CLS 2
    LOCATE 10, 20: PRINT "TRANSMIT LOOP RESPONSE VALUE = ";
RESPONSE&
    LOCATE 14, 20: PRINT "OR PRESS "; RET$; " RETURN FOR MAIN MENU"
    LOCATE 12, 20: INPUT ; "ENTER A NEW VALUE: ", RESP&
    SELECT CASE RESP&
        CASE 1 TO 1000000: RESPONSE& = RESP&: GOTO NewResp
        CASE ELSE: GOTO MainMenu
    END SELECT
NewResp:
    RESPFILNUM = FREEFILE
    OPEN "RESP.DAT" FOR OUTPUT AS #RESPFILNUM
    PRINT #RESPFILNUM, RESPONSE&: CLOSE #RESPFILNUM
    GOTO MainMenu
'

```

```

*****
'*   Z.   ZERO SPEED LOOP TIMEOUT VALUE                               *
*****

```

ZERORESP:

```

    FILLER$ = " ZERO SPD LOOP RESPONSE"
    GOSUB BottomLine25: COLOR 7, 0: CLS 2
    LOCATE 10, 20: PRINT "LOOP RESPONSE VALUE = "; ZERORESP&
    LOCATE 14, 20: PRINT "OR PRESS "; RET$; " RETURN FOR MAIN MENU"
    LOCATE 12, 20: INPUT ; "ENTER A NEW VALUE: ", ZRESP&
    SELECT CASE ZRESP&
        CASE 1 TO 1000000000: ZERORESP& = ZRESP&: GOTO ZNewResp
        CASE ELSE: GOTO MainMenu
    END SELECT

```

ZNewResp:

```

    ZRESPFILNUM = FREEFILE
    OPEN "ZRESP.DAT" FOR OUTPUT AS #ZRESPFILNUM
    PRINT #ZRESPFILNUM, ZERORESP&: CLOSE #ZRESPFILNUM
    GOTO MainMenu

```

```

*****
'*   O.   ON COM TIMEOUT RESPONSE VALUE                               *
*****

```

OnComResp:

```

    FILLER$ = " ON COM TIMEOUT      "
    GOSUB BottomLine25: COLOR 7, 0: CLS 2
    LOCATE 10, 20: PRINT "TIMEOUT VALUE = "; CRESPONSE&
    LOCATE 14, 20: PRINT "OR PRESS "; RET$; " RETURN FOR MAIN MENU"
    LOCATE 12, 20: INPUT ; "ENTER A NEW VALUE: ", CRESP&
    SELECT CASE CRESP&
        CASE 1 TO 1000000: CRESPONSE& = CRESP&: GOTO CNewResp
        CASE ELSE: GOTO MainMenu
    END SELECT

```

CNewResp:

```

    CRESPFILNUM = FREEFILE
    OPEN "CRESP.DAT" FOR OUTPUT AS #CRESPFILNUM
    PRINT #CRESPFILNUM, CRESPONSE&: CLOSE #CRESPFILNUM
    GOTO MainMenu

```

```

*****
'*   C.   CONFIGURE BAUD & COMPORT                                   *
*****

```

ChangeBaudCom:

```

    FILLER$ = " CONFIG BAUD & COMPORT "

```

```

GOSUB BottomLine25: COLOR 7, 0: CLS 2
CLOSE #1
GetBaud:
OPEN "ABAUD.DAT" FOR INPUT AS #1: INPUT #1, A$: CLOSE #1
LOCATE 10, 30: PRINT "BAUD RATE "; A$
LOCATE 13, 30: PRINT "RETURN TO ENTER"
LOCATE 10, 42: LINE INPUT BAUD$
SELECT CASE BAUD$
CASE "": CLS 2: BAUD$ = A$: GOTO GetCom
CASE "300": BAUDBOT$ = " 300": GOTO GoodBaud
CASE "600": BAUDBOT$ = " 600": GOTO GoodBaud
CASE "1200": BAUDBOT$ = " 1200": GOTO GoodBaud
CASE "2400": BAUDBOT$ = " 2400": GOTO GoodBaud
CASE "4800": BAUDBOT$ = " 4800": GOTO GoodBaud
CASE "9600": BAUDBOT$ = " 9600": GOTO GoodBaud
CASE "19200": BAUDBOT$ = "19200": GOTO GoodBaud
CASE ELSE: GOTO ChangeBaudCom
END SELECT
GoodBaud:
OPEN "ABAUD.DAT" FOR OUTPUT AS #1
PRINT #1, BAUD$: CLOSE #1
GetCom:
OPEN "ACOM.DAT" FOR INPUT AS #1: INPUT #1, A$: CLOSE #1
GOSUB BottomLine25: COLOR 7, 0: CLS 2
LOCATE 10, 30: PRINT "COM1 OR COM2 "; A$
LOCATE 13, 30: PRINT "RETURN TO ENTER"
LOCATE 10, 43: LINE INPUT MCOM$
SELECT CASE MCOM$
CASE "": MCOM$ = A$: GOTO MainMenu
CASE "1": GOTO GoodCom
CASE "2": GOTO GoodCom
CASE ELSE: GOTO GetCom
END SELECT
GoodCom:
OPEN "ACOM.DAT" FOR OUTPUT AS #1
PRINT #1, MCOM$: CLOSE #1
GOTO MainMenu
,
'*****
'*      S.      SHELL TO A PROGRAM      *
'*****
,
ShellToProg:
SFLAG = 1 '** SET SHELL FLAG FOR ON ERROR OPEN CHECK
SHNM$ = ""
FILENUM = FREEFILE

```

```

PRINT "ENTER THE NAME OF THE FILE"
INPUT "(e.g. DEG3.DAT) "; NAMES$
PRINT
OPEN NAMES$ FOR INPUT AS #1
INPUT #1, POINTS
FOR K = 1 TO POINTS
    INPUT #1, POST(K)
NEXT K
TINC = .01
DTINC = .05
TMAX = .05 * (POINTS - 1)
POSTIME(1) = POST(1)
I = 1
J = 1
1  IF (TINC * J) > TMAX GOTO 2
3  IF (TINC * J) > (DTINC * I) GOTO 4
    SLOPE = (POST(I + 1) - POST(I)) / DTINC
    POSTIME(J + 1) = POST(I) + SLOPE * (TINC * J - DTINC * (I - 1))
    J = J + 1
    GOTO 1
4  I = I + 1
    GOTO 3
2  DATAFLAG = 1
    POINTS = J - 1
    CLOSE #1
REM *****
REM COMMENT OUT THE FOLLOWING IF ACTUAL 0.01 INC. DATA FILE IS
REM NOT REQUIRED
REM
    OPEN "POINT.DAT" FOR OUTPUT AS #1
    FOR I = 1 TO POINTS STEP 1
        PRINT #1, POSTIME(I)
    NEXT I
REM *****
    CLOSE #1
    CLS
    PRINT : PRINT : PRINT : PRINT : PRINT
    PRINT "          HIT RETURN KEY TO CONTINUE "
    CONTINUE$ = INPUT$(1)
    PRINT

    GOTO 400
REM *****
REM          TRANSIENT OPTION #3: RUN XTGOLD
REM *****

```

```

480  CLS
REM  TO EDIT THE FILE USE XTPRO GOLD EDITOR
      SHELL "XTG"
      GOTO 400

REM  *****
REM      TRANSIENT OPTION #4: RUN EXISTING DATA
REM  *****
500  CLS
REM  THIS SECTION WILL RUN A CREATED DATA FILE. THE VALVE
REM  IS INITIALIZED TO THE FIRST DATA POINT IN THE FILE. TRSTEP
REM  WAS INITIALIZED AT THE BEGINNING OF THE PROGRAM.
REM
REM
      IF DATAFLAG = 0 THEN 510 ELSE 520
510  CLS
      LOCATE 10, 20: PRINT " NO DATA IN VALVE POSITION ARRAY "
      FOR L = 1 TO 20000
      NEXT L
      GOTO 400

520  CLS
      LOCATE 10, 20: PRINT " **** MOVING VALVE TO INITIAL STARTING
      POSITION *** "
REM  FOR TRANSIENT VALVE OPERATION SET MASSCOMP TRIGGER TO
REM  ZERO
      GOSUB TRIGLOW
      NEWANGLE = POSTIME(1)
      GOSUB MOVEVALVE

      CLS
REM
REM  OPEN SERIAL PORT COM2 TO THE MOTOR PC.
      OPEN "COM2:9600,N,7,1" FOR OUTPUT AS #2

      LOCATE 10, 20: PRINT " HIT RETURN KEY TO START TRANSIENT
      VALVE MOVEMENT "
      CONTINUE$ = INPUT$(1)
      PRINT
REM  SET MASSCOMP TRIGGER HIGH TO START DATA ACQUISITION
      GOSUB TRIGHIGH
      LOCATE 12, 20: PRINT " MASSCOMP HAS BEEN TRIGGERED "

REM  PUT A SHORT DELAY BETWEEN START OF DATA ACQUISITION AND
REM  MOVEMENT OF VALVE
REM  SET SHORT = 1000 FOR NORMAL OPERATION

```



```

REM SET SHORT = 10000 FOR 1.5 SECOND DELAY FOR HIGH SPEED
REM CAMERA TO GET TO SPEED FOR 500 FPS FRAME RATE
REM (ALSO SET PORTION BELOW FOR CAMERA OPERATION)

    FOR SHORT = 1 TO 10000
    NEXT SHORT
    LOCATE 15, 20: PRINT " *** VALVE MOVING **** "
REM TRIGGER MOTOR CONTROLLER BY SENDING THE LETTER "A" OVER
SERIAL
REM COM #2 TO THE MOTOR PC.
    PRINT #2, "A"
REM
REM FOR AN IBM AT AND 0.01 SECOND INTERVALS SET TRSTEP = 80
REM TRSTEP WAS SET AT THE BEGINNING OF THIS PROGRAM.
REM
    FOR K = 1 TO POINTS STEP 1
    VOLTS = VSLOPE * POSTIME(K) + VCLOSED
    DIGITAL = INT(DSLOPE * VOLTS + DLOW)
        DH% = INT(DIGITAL / 16!)
        DL% = DIGITAL - 16! * DH%
        DL% = 16! * DL%
        OUT &H340, DL%
        OUT &H341, DH%
        FOR SLOW = 1 TO TRSTEP
        NEXT SLOW
    NEXT K
    OLDANGLE = POSTIME(POINTS)
    CLS
    LOCATE 10, 20: PRINT " *** FINISHED MOVING VALVE ****"

REM PUT A SHORT DELAY HERE TO ALLOW MASSCOMP COMPUTER TO
REM FINISH. THIS ALLOWS THE MASSCOMP TO STOP ACQUIRING DATA
REM WHEN THE CLOCK DICTATES (@ 5 SECONDS) RATHER THAN THE
REM TRIGGER

REM SET SHORT = 1 TO 10000 FOR NORMAL OPERATION
REM SET SHORT = 1 TO 9000 FOR CAMERA OPERATION WITH 5 SECOND
REM RUN AND 500 FPS FRAME RATE

    FOR SHORT = 1 TO 12000
    NEXT SHORT
    GOSUB TRIGLOW
    CLOSE #2
    GOTO 400

REM

```

```

REM *****
REM  TRANSIENT OPTION #5: CLOSE VALVE (SET TO 0 DEGREES)
REM *****
560  CLS

      NEWANGLE = 0!
      LOCATE 10, 20: PRINT " >>>>>> CLOSING VALVE <<<<<<"
      GOSUB MOVEVALVE
      LOCATE 20, 20: PRINT " >>>>>> VALVE CLOSED <<<<<<"
      FOR SHORT = 1 TO 12000
      NEXT SHORT
      GOTO 400

REM *****
600  PRINT " END OF PROGAM "
END
REM *****
REM
REM
REM          SUBROUTINES
REM
REM
REM *****
REM          STEADY STATE MOVEMENT OF VALVE
REM *****

```

MOVEVALVE:

```

      LOCATE 15, 20: PRINT " *****  MOVING VALVE *****"

      IF NEWANGLE = OLDANGLE THEN 800
      IF NEWANGLE < OLDANGLE THEN KANGLE = -STEPANGLE
      IF NEWANGLE > OLDANGLE THEN KANGLE = STEPANGLE

      FOR VALVEANGLE = OLDANGLE TO NEWANGLE STEP KANGLE
      VOLTS = VSLOPE * VALVEANGLE + VCLOSED
      DIGITAL = INT(DSLOPE * VOLTS + DLOW)
          DH% = INT(DIGITAL / 16!)
          DL% = DIGITAL - 16! * DH%
          DL% = 16! * DL%
          OUT &H340, DL%
          OUT &H341, DH%
          SHOWANGLE = INT(VALVEANGLE * 100) / 100!
      LOCATE 18, 20: PRINT " CURRENT VALVE POSITION : "; SHOWANGLE
      LOCATE 18, 55: PRINT "( degrees )"
      FOR J = 1 TO STDSTEP

```

```

        NEXT J
        NEXT VALVEANGLE
800    OLDANGLE = NEWANGLE

```

```

RETURN

```

```

REM *****

```

```

REM          MASSCOMP HIGH TRIGGER

```

```

REM *****

```

```

TRIGHIGH:

```

```

    DL% = 0

```

```

    DH% = 125

```

```

    OUT &H342, DL%

```

```

    OUT &H343, DH%

```

```

    RETURN

```

```

REM *****

```

```

REM          MASSCOMP LOW TRIGGER

```

```

REM *****

```

```

TRIGLOW:

```

```

    DL% = 0

```

```

    DH% = 0

```

```

    OUT &H342, DL%

```

```

    OUT &H343, DH%

```

```

    RETURN

```

```

OPEN "SHELNAME.DAT" FOR INPUT AS #FILENUM
INPUT #FILENUM, SHNM$: CLOSE #FILENUM
ShellToProg1:
  IF SFLAG = 2 THEN SHNM$ = "NONE" '** SHELNAME.DAT FILE DOESN'T
  EXIST
  FILLER$ = " SHELL TO A PROGRAM  "
  GOSUB BottomLine25: COLOR 7, 0: CLS 2
  LOCATE 4, 30: PRINT "SHELL TO A PROGRAM";
  LOCATE 7, 10
  PRINT "YOU MUST ENTER THE FULL PATH OF THE PROGRAM YOU WISH
  TO RUN";
  LOCATE 9, 12
  PRINT "e.g.: TO RUN PROCOMM FROM ANOTHER DRIVE OR
  DIRECTORY...";
  LOCATE 11, 18: PRINT "ENTER: A:\PROCOMM or C:\MODEMPROCOMM";
  LOCATE 14, 10: PRINT "PRESS " + RET$ + " TO USE THE DEFAULT OF: ";
  SHNM$
  LOCATE 16, 10: INPUT ; "ENTER PROGRAM NAME: ", A$
  IF A$ = "" AND SHNM$ = "NONE" THEN
    LOCATE 21, 10
    PRINT "PRESS 'ESC' FOR MAIN MENU OR " + RET$ + " TO TRY AGAIN";
ShellToProg2:
  E$ = INKEY$
  SELECT CASE E$
    CASE CHR$(13): GOTO ShellToProg1
    CASE CHR$(27): GOTO MainMenu
    CASE ELSE: GOTO ShellToProg2
  END SELECT
END IF
LOCATE 20, 8: PRINT "'ESC' - TO ABORT TO MAIN MENU :";
LOCATE 21, 9: PRINT RET$ + " - TO RUN PROGRAM ENTERED:";
LOCATE 22, 8: PRINT "'S' - TO SAVE AS NEW DEFAULT:";
IF A$ = "" THEN GOTO DoShellOptions ELSE SHNM$ = A$
DoShellOptions:
  A$ = INKEY$
  LOCATE 21, 40, 1
  SELECT CASE A$
    CASE CHR$(27): GOTO MainMenu
    CASE CHR$(13): GOTO RunShellName
    CASE "S", "s": GOTO SaveShellName
    CASE ELSE: GOTO DoShellOptions
  END SELECT
RunShellName:
  CLS : LOCATE 20, 1: SHELL SHNM$** RUN PROGRAM
  LOCATE 25, 1: PRINT STRING$(80, 32);
  LOCATE 25, 30: PRINT "ANY KEY TO CONTINUE...";

```

RunShellName1:

A\$ = INKEY\$: IF A\$ = "" THEN GOTO RunShellName1

GOTO MainMenu

SaveShellName:

FILENUM = FREEFILE

OPEN "SHELNAME.DAT" FOR OUTPUT AS #FILENUM

PRINT #FILENUM, SHNM\$: CLOSE #FILENUM

LOCATE 22, 40: PRINT "SAVING: "; SHNM\$;

FOR X = 1 TO 30000: NEXT

LOCATE 22, 40: PRINT STRING\$(35, 32);

GOTO DoShellOptions

* Q. QUIT TO DOS *

QuitToDOS:

CLS 2: CLOSE : SYSTEM

* ON ERROR TRAPPING ROUTINES *

ErrorTrap:

BEEP

IF ERR = 57 THEN

CLS 2

LOCATE 15, 25: PRINT "COM PORT NOT RESPONDING CHECK CABLE"

LOCATE 16, 25: PRINT "PRESS ANY KEY FOR MAIN MENU"

GOTO ExitErrorTrap

END IF

IF ERR = 25 THEN

CLS 2

LOCATE 15, 25: PRINT "PRINTER NOT RESPONDING"

LOCATE 16, 25: PRINT "PRESS ANY KEY FOR MAIN MENU"

GOTO ExitErrorTrap

END IF

IF ERR = 53 THEN

IF SFLAG = 1 THEN SFLAG = 2: CLOSE #FILENUM: RESUME ShellToProg1

IF DFLAG = 0 THEN

CLS 2

LOCATE 6, 10

PRINT "ONE OR MORE OF THE FOLLOWING FILES DO NOT EXIST:"

LOCATE 8, 10

PRINT "ABAUD.DAT - ACOM.DAT - RESP.DAT - CRESP.DAT -

ZRESP.DAT"

```

LOCATE 14, 10
PRINT "PRESS " + RET$;
PRINT " TO CREATE THEM USING DEFAULTS OF:"
PRINT TAB(10); "19200 BAUD, COM PORT #1, TRANSMIT LOOP
RESP=100, ";
PRINT TAB(10); "ON COM(n) RESP=100, ZERO SPEED LOOP RESP=10000"
LOCATE 18, 10
PRINT "AFTER WHICH YOU MAY CHANGE THE DEFAULTS FROM THE
MAIN MENU."
LOCATE 20, 10
PRINT "OR PRESS 'ESC' TO EXIT THIS PROGRAM."
A$ = INPUT$(1)
IF A$ = CHR$(27) THEN CLS 2: CLOSE : SYSTEM
IF A$ = CHR$(13) THEN
    CLOSE : BAUD$ = "19200": MCOM$ = "1"
    RESPONSE& = 100: CRESPONSE& = 100: ZERORESP& = 10000
    OPEN "ABAUD.DAT" FOR OUTPUT AS #1: PRINT #1, BAUD$: CLOSE #1
    OPEN "ACOM.DAT" FOR OUTPUT AS #1: PRINT #1, MCOM$: CLOSE 1
    OPEN "RESP.DAT" FOR OUTPUT AS #1: PRINT #1, RESPONSE&:
    CLOSE #1
    OPEN "CRESP.DAT" FOR OUTPUT AS #1: PRINT #1, CRESPONSE&:
    CLOSE #1
    OPEN "ZRESP.DAT" FOR OUTPUT AS #1: PRINT #1, ZERORESP&:
    CLOSE #1
    END IF
    RESUME
ELSE
    CLS 2
    LOCATE 12, 10: PRINT "FILE DOES NOT EXIST....."
    LOCATE 16, 10: PRINT "PRESS ANY KEY FOR MAIN MENU"
    GOTO ExitErrorTrap
END IF
END IF
IF ERR = 64 OR ERR = 75 THEN
    CLS 2: LOCATE 15, 10
    PRINT "ILLEGAL FILE NAME - PRESS ANY KEY FOR MAIN MEMU"
    GOTO ExitErrorTrap
END IF
CLS 2
LOCATE 15, 10
PRINT "ERROR NUMBER "; ERR; " PRESS ANY KEY FOR MAIN MENU"
ExitErrorTrap:
A$ = INKEY$: IF A$ = "" THEN GOTO ExitErrorTrap ELSE GOTO MainMenu

```

```

*****
'*          BEGIN ALL SUBROUTINES          *
*****

'**** SUB TO CALCULATE CHECKSUM FOR WRITE PACKETS ****
'
CalcChksumWrite:
  CS = 3
  FOR CNT = 1 TO 3
    CS = CS XOR ASC(MID$(P$, CNT, 1)): NEXT
  FOR CNT = 1 TO 4
    CS = CS XOR ASC(MID$(V$, CNT, 1)): NEXT
  IF CS < 32 THEN CS = CS + 32
  RETURN
'
'**** END SUB
*****

'**** SUB TO CALCULATE CHECKSUM FOR READ PACKETS ****
'
CalcChksumRead:
  CS = 0
  FOR CNT = 1 TO 8
    CS = CS XOR ASC(MID$(A$, CNT, 1)): NEXT CNT
  IF CS < 32 THEN CS = CS + 32
  RETURN
'
'**** END SUB
*****

'**** SUB TO PAD V$ (PARAMETER VALUE) TO 4 CHARACTERS ****
'
PADV:
  IF V < 0 THEN S$ = "-" ELSE S$ = "+"
  IF S$ = "-" THEN V = V * (-1) '** MAKE V POSITIVE
  V$ = STR$(V)
  '** V$ HAS LEADING 'SPACE' WHICH MUST BE CLEARED
  V$ = RIGHT$(V$, (LEN(V$) - 1))
  SELECT CASE LEN(V$)
    CASE 1: V$ = S$ + "00" + V$
    CASE 2: V$ = S$ + "0" + V$
    CASE 3: V$ = S$ + V$
    CASE ELSE: RETURN

```

```

END SELECT
RETURN
,
'***** END SUB *****
,
,
'***** SUB TO PAD P$ (PARAMETER NUMBER) TO 3 CHARACTERS *****
PADP:
P$ = STR$(P)
'***** P$ HAS LEADING 'SPACE' WHICH MUST BE CLEARED
P$ = RIGHT$(P$, (LEN(P$) - 1))
SELECT CASE LEN(P$)
    CASE 1: P$ = "00" + P$
    CASE 2: P$ = "0" + P$
    CASE ELSE: RETURN
END SELECT
RETURN
,
'***** END SUB *****
,
,
'***** SUB TO TURN ON COM PORT OR CLOSE AND REOPEN COM PORT
,
TurnOnCom:
'** CLEAR COM PORT BUFFER & OPEN COM PORT
CLOSE
OPEN COMFIL$ FOR RANDOM AS #1
ON COM(VAL(MCOM$)) GOSUB ComDetect
COM(VAL(MCOM$)) ON
RETURN
,
'***** END SUB *****
,
,
'***** SUB ON COM(n) TO DETECT CHARACTERS RECEIVED AT COM(n)
    PORT *****
,
ComDetect:
RX = 1 ' SET Receive Flag
RETURN
,
'***** END SUB
*****
,
,

```


'***** SUB TO READ ONE READ PACKET FROM DRIVE *****

ReadPacket:

RX = 0

GOODREAD = 0

IF SCROLLING = 1 THEN

PRINT #1, O\$;

ELSE

PRINT #1, CHR\$(4); ADMSB\$; ADMSB\$; ADLSB\$; ADLSB\$; P\$; CHR\$(5);

END IF

** WAIT FOR ON COM(n) SUB ComDetect TO RESPOND **

FOR COUNT& = 1 TO CRESPONSE&: NEXT COUNT&

IF RX = 1 THEN ** CHARACTERS HAVE BEEN RECEIVED AT COM(n) PORT
**

A\$ = INPUT\$(1, #1)

IF ASC(A\$) = 4 THEN

V\$ = "Parameter is not used"

RETURN

END IF

IF ASC(A\$) = 2 THEN ** VALUE PACKET RECEIVED

A\$ = INPUT\$(9, #1)

GOSUB CalcChksumRead

IF CS = ASC(RIGHT\$(A\$, 1)) THEN

GOODREAD = 1

P\$ = LEFT\$(A\$, 3): V\$ = MID\$(A\$, 4, 4)

RETURN

ELSE

V\$ = "Checksum ERROR.. Try Again..."

RETURN

END IF

ELSE

V\$ = "Incorrect Drive Response"

END IF

RETURN

ELSE

V\$ = " COM Port Timeout: Drive NOT responding to Read... "

END IF

RETURN

'***** END SUB *****

'***** SUB TO UPLOAD ONE WRITE PACKET TO DRIVE *****

WritePacket:

RX = 0 'RESET RECEIVE FLAG

```

PRINT #1, CHR$(4); ADMSB$; ADMSB$; ADLSB$; ADLSB$; CHR$(2);
PRINT #1, P$; V$; CHR$(3); CHR$(CS);
** WAIT FOR ON COM(n) SUB ComDetect TO RESPOND **
FOR COUNT& = 1 TO CRESPONSE&: NEXT COUNT&
IF RX = 1 THEN ** CHARACTERS HAVE BEEN RECEIVED AT COM(n) PORT
**
    A$ = INPUT$(1, #1)
    IF ASC(A$) = 6 THEN PRINT "Successful Write": RETURN
    IF ASC(A$) = 21 THEN PRINT "RO, Bad Data or Checksum Error": RETURN
    PRINT "Incorrect Drive Response"
    RETURN
ELSE
    PRINT " COM Port Timeout: Drive NOT responding to Write... "
END IF
RETURN
,
***** END SUB
*****
,
***** SUB USED FOR TIMING OR PAUSE *****
,
TimeLoop:
    FOR T& = 1 TO RESPONSE&: NEXT
    RETURN
,
***** END SUB *****
,
***** SUB TO DISPLAY BOTTOM LINE 25 OF SCREEN *****
,
BottomLine25:
    LOCATE 25, 1
    COLOR 0, 7
    PRINT FILLER$; SPC(2); CHR$(221); SPC(1);
    PRINT "DRIVE #: "; AD$; SPC(2); CHR$(221); SPC(1);
    PRINT "PORT: COM" + MCOM$; SPC(2); CHR$(221); SPC(1);
    PRINT "BAUD RATE: " + BAUDBOT$ + " " + PARITYDATASTOP$; SPC(2);
    RETURN
,
***** END SUB
*****

```

The following program is VCONIMPL.BAS, which controls the triggering of the Impeller Test Facility as well as the movement of the control valve.

```
REM  VCONIMPL.BAS
REM
REM
*****
REM  VALVE CONTROL PROGRAM FOR IMPELLER TEST FACILITY
REM
*****
REM  COMMENTS:
REM
REM  THE INPUT DATA FILE FOR VALVE POSITION IS AT TIME
REM  INCREMENTS OF 0.05 SECONDS. THIS PROGRAM THEN LINEARLY
REM  INTERPOLATES THAT DATA TO CREATE AN ARRAY OF VALVE
REM  POSITION AT THE 0.01 SECOND INCREMENT FOR WHICH THE
REM  CONTROL PORTION OF THIS PROGRAM IS EXECUTED.
REM  NOTE: THE NUMBER OF VALVE POSITION DATA POINTS IN THE
REM  FILE MUST BE LISTED IN THE FIRST LINE OF THE DATA FILE.
REM
REM  IF MORE THAN 100 POINTS ARE IN THE DATA FILE, CHANGE THE
REM  DIMENSION STATEMENT.
REM
REM  THE TIME STEP DURING A TRANSIENT IS CONTROLLED BY THE
REM  VARIABLE CALLED "TRSTEP" WHICH IS DEFINED AT THE
REM  BEGINNING OF THE PROGRAM.
REM
REM  THE TIME STEP USED TO CHANGE VALVE POSITION FOR STEADY
REM  STATE OPERATION IS CONTROLLED BY THE VARIABLE CALLED
REM  "STDSTEP" AND "STEPANGLE" BOTH DEFINED AT THE BEGINNING
REM  OF THE PROGRAM.
REM
REM  OUTPUT IS NEGATIVE TO THE MOOG SERVO ELECTRONICS:
REM  0 (V) CORRESPONDS WITH 2048 (DEC. EQ.) AND
REM  -10 (V) CORRESPONDS WITH 4095 (DEC. EQ.)
REM
REM  THE CODE IS WRITTEN FOR THE DAC-02 D/A BOARD.
REM
REM
REM  BASE I/O ADDRESS = HEX 340.
REM  VALVE CONTROL OUTPUT
REM  OUTPUT = PIN 23, JUMPER = PIN 20 & 22, GROUND = PIN 2.
REM
REM  MASSCOMP TRIGGER OUTPUT
REM  OUTPUT = PIN 18, JUMPER = PIN 15 & 16 , GROUND = PIN 2.
```

```

REM
REM  MOTOR CONTROLLER TRIGGERED VIA OUTPUT OF LETTER "A" ON
REM  SERIAL COM2.
REM
REM *****
      DIM POSTIME(10000), POST(100)
REM
REM ***** PUT OPEN AND CLOSED VALVE VOLTAGE LEVELS HERE *****
REM
REM  VALVE POSITION    DEGREES    VOLTS
REM  -----
REM    CLOSED         0    VCLOSED
REM    OPEN           90    VOPEN
REM
      VOPEN = 9.3
      VCLOSED = 1.61
      VRANGE = VOPEN - VCLOSED
      VSLOPE = VRANGE / 90!
REM *****
REM  TRSTEP CONTROLS THE TRANSIENT TIME STEP.
REM  FOR AN IBM-AT AND 0.01 SECOND INTERVALS SET "TRSTEP = 80".
REM  STDSTEP CONTROLS THE STEADY STATE TIME STEP.
REM  FOR AN IBM-AT AND ABOUT 10 SECOND TO CHANGE 90 DEGREES,
REM  SET STDSTEP = 80 AND STEPANGLE = 0.1
REM  THE ABOVE VALUES WILL CHANGE IF RUN IN THE QB
REM  ENVIRONMENT.
REM *****
      TRSTEP = 80!
      STDSTEP = 80!
      STEPANGLE = .1
REM *****
      DHIGH = 4095
      DLOW = 2048
      DRANGE = DHIGH - DLOW
      DSLOPE = DRANGE / 10!
100  COLOR 15, 1
      CLS
REM *****
REM  AT START OF PROGRAM PUT VALVE IN CLOSED POSITION
REM *****
      DATAFLAG = 0
      NEWANGLE = 0!
      OLDANGLE = 90!

      PRINT : PRINT : PRINT : PRINT : PRINT : PRINT : PRINT : PRINT
      PRINT "          VALVE WILL INITIALLY CLOSE."

```

```

      GOSUB MOVEVALVE
REM
REM
REM *****
120  CLS : PRINT : PRINT : PRINT : PRINT
      PRINT "                MAIN MENU"
      PRINT "                -----"
      PRINT : PRINT
      PRINT "                PLEASE SELECT ONE OF THE FOLLOWING:"
      PRINT
      PRINT "                (S) Steady state valve operation."
      PRINT
      PRINT "                (T) Transient valve operation."
      PRINT
      PRINT "                (E) Exit to DOS."
      PRINT : PRINT : PRINT
      LOCATE 18, 24: OPTION$ = INPUT$(1)
            IF OPTION$ = "S" OR OPTION$ = "s" THEN 200
            IF OPTION$ = "T" OR OPTION$ = "t" THEN 400
            IF OPTION$ = "E" OR OPTION$ = "e" THEN 600
      GOTO 120
REM
REM *****
200  PRINT " ***** STEADY STATE PORTION OF
      PROGRAM ****"
REM *****
      COLOR 15, 1
      CLS
REM  FOR STEADY STATE OPERATION OF VALVE SET MASSCOMP TRIGGER
REM  TO A HIGH VALUE DURING OPERATION
      GOSUB TRIGHIGH
REM
      PRINT : PRINT
      PRINT "                COMPUTER CONTROLLED VALVE POSITIONER"
      PRINT "                -----"
      PRINT : PRINT
      PRINT "                CLOSED POSITION  = 0 DEG"
      PRINT "                OPEN POSITION   = 90 DEG"
      PRINT
      PRINT "                PRESENT POSITION = "; OLDANGLE; " DEG "
      PRINT : PRINT : PRINT
      PRINT "    DO YOU WISH TO SELECT A NEW VALVE POSITION (Y/N) "
      OPTION$ = INPUT$(1)
            IF OPTION$ = "Y" OR OPTION$ = "y" THEN 220
            IF OPTION$ = "N" OR OPTION$ = "n" THEN 240
      GOTO 200

```

```

REM
220 PRINT : PRINT
    INPUT "      ENTER DESIRED VALVE POSITION IN DEGREES "; PS
    IF PS < 0! OR PS > 90! THEN 220
    NEWANGLE = PS
        CLS
        GOSUB MOVEVALVE
    GOTO 200

REM ----- OPTIONS MENU -----

240 CLS
    PRINT : PRINT : PRINT
    PRINT "          *** OPTIONS MENU ***"
    PRINT : PRINT
    PRINT "          (V) VALVE POSITION MENU"
    PRINT
    PRINT "          (M) MAIN MENU"
    PRINT
    PRINT : PRINT : PRINT
    PRINT "          WHAT IS YOUR SELECTION "
    OPTION$ = INPUT$(1)
        IF OPTION$ = "V" OR OPTION$ = "v" THEN 200
        IF OPTION$ = "M" OR OPTION$ = "m" THEN 120
    GOTO 240
REM *****

400 CLS

REM
PRINT : PRINT
PRINT "          TRANSIENT VALVE OPERATION"
PRINT "          -----"
PRINT
PRINT "          PLEASE SELECT ONE OF THE FOLLOWING:"
PRINT
PRINT "          (V) Validate time step."
PRINT
PRINT "          (G) Get valve position data from file."
PRINT
PRINT "          (E) Use XTGOLD to edit data file "
PRINT
PRINT "          (R) Run existing data."
PRINT
PRINT "          (C) Close valve."

```

```

PRINT
PRINT "          (M) MAIN MENU."
PRINT
  LOCATE 23, 23: OPTION$ = INPUT$(1)
  IF OPTION$ = "V" OR OPTION$ = "v" THEN 440
  IF OPTION$ = "G" OR OPTION$ = "g" THEN 460
  IF OPTION$ = "E" OR OPTION$ = "e" THEN 480
  IF OPTION$ = "R" OR OPTION$ = "r" THEN 500
  IF OPTION$ = "C" OR OPTION$ = "c" THEN 560
  IF OPTION$ = "M" OR OPTION$ = "m" THEN 120
  GOTO 400
REM *****
REM      TRANSIENT OPTION #1: VALIDATE TRANSIENT TIME STEP
REM *****
440  CLS
    LOCATE 10, 20: PRINT "**** SHUT OFF HYDRAULIC PUMP ****"
    LOCATE 12, 20: PRINT "CONNECT COMMAND OUTPUT TO SCOPE"
    FOR K = 1 TO 9999 STEP 2
      POSTIME(K) = 0!
      POSTIME(K + 1) = 90!
    NEXT K
REM
    PRINT "PRESS ANY KEY TO START TIME STEP VALIDATION"
REM
    BEGIN$ = INPUT$(1)
    FOR J = 1 TO 5000 STEP 1
      VOLTS = VSLOPE * POSTIME(J) + VCLOSED
      DIGITAL = INT(DSLOPE * VOLTS + DLOW)
      DH% = INT(DIGITAL / 16!)
      DL% = DIGITAL - 16! * DH%
      DL% = 16! * DL%
      OUT &H340, DL%
      OUT &H341, DH%
      FOR SLOW = 1 TO TRSTEP
        NEXT SLOW
    NEXT J
    FOR K = 1 TO 9999 STEP 1
      POSTIME(K) = 90!
      POSTIME(K + 1) = 90!
    NEXT K
    DATAFLAG = 0
    GOTO 400
REM *****
REM      TRANSIENT OPTION #2: GET A DATA FILE
REM *****
460  CLS

```

```

PRINT "ENTER THE NAME OF THE FILE"
INPUT "(e.g. DEG3.DAT) "; NAMES$
PRINT
OPEN NAMES$ FOR INPUT AS #1
INPUT #1, POINTS
FOR K = 1 TO POINTS
    INPUT #1, POST(K)
NEXT K
TINC = .01
DTINC = .05
TMAX = .05 * (POINTS - 1)
POSTIME(1) = POST(1)
I = 1
J = 1
1  IF (TINC * J) > TMAX GOTO 2
3  IF (TINC * J) > (DTINC * I) GOTO 4
    SLOPE = (POST(I + 1) - POST(I)) / DTINC
    POSTIME(J + 1) = POST(I) + SLOPE * (TINC * J - DTINC * (I - 1))
    J = J + 1
    GOTO 1
4  I = I + 1
    GOTO 3
2  DATAFLAG = 1
    POINTS = J - 1
    CLOSE #1
REM *****
REM COMMENT OUT THE FOLLOWING IF ACTUAL 0.01 INC. DATA FILE IS
REM NOT REQUIRED
REM
    OPEN "POINT.DAT" FOR OUTPUT AS #1
    FOR I = 1 TO POINTS STEP 1
        PRINT #1, POSTIME(I)
    NEXT I
REM *****
    CLOSE #1
    CLS
    PRINT : PRINT : PRINT : PRINT : PRINT
    PRINT "          HIT RETURN KEY TO CONTINUE "
    CONTINUE$ = INPUT$(1)
    PRINT

    GOTO 400
REM *****
REM          TRANSIENT OPTION #3: RUN XTGOLD
REM *****

```



```

480  CLS
REM  TO EDIT THE FILE USE XTPRO GOLD EDITOR
      SHELL "XTG"
      GOTO 400

REM  *****
REM      TRANSIENT OPTION #4: RUN EXISTING DATA
REM  *****
500  CLS
REM  THIS SECTION WILL RUN A CREATED DATA FILE. THE VALVE
REM  IS INITIALIZED TO THE FIRST DATA POINT IN THE FILE. TRSTEP
REM  WAS INITIALIZED AT THE BEGINNING OF THE PROGRAM.
REM
REM
      IF DATAFLAG = 0 THEN 510 ELSE 520
510  CLS
      LOCATE 10, 20: PRINT " NO DATA IN VALVE POSITION ARRAY "
      FOR L = 1 TO 20000
      NEXT L
      GOTO 400

520  CLS
      LOCATE 10, 20: PRINT " **** MOVING VALVE TO INITIAL STARTING
      POSITION *** "
REM  FOR TRANSIENT VALVE OPERATION SET MASSCOMP TRIGGER TO
REM  ZERO
      GOSUB TRIGLOW
      NEWANGLE = POSTIME(1)
      GOSUB MOVEVALVE

      CLS
REM
REM  OPEN SERIAL PORT COM2 TO THE MOTOR PC.
      OPEN "COM2:9600,N,7,1" FOR OUTPUT AS #2

      LOCATE 10, 20: PRINT " HIT RETURN KEY TO START TRANSIENT
      VALVE MOVEMENT "
      CONTINUE$ = INPUT$(1)
      PRINT
REM  SET MASSCOMP TRIGGER HIGH TO START DATA ACQUISITION
      GOSUB TRIGHIGH
      LOCATE 12, 20: PRINT " MASSCOMP HAS BEEN TRIGGERED "

REM  PUT A SHORT DELAY BETWEEN START OF DATA ACQUISITION AND
REM  MOVEMENT OF VALVE
REM  SET SHORT = 1000 FOR NORMAL OPERATION

```

```

REM SET SHORT = 10000 FOR 1.5 SECOND DELAY FOR HIGH SPEED
REM CAMERA TO GET TO SPEED FOR 500 FPS FRAME RATE
REM (ALSO SET PORTION BELOW FOR CAMERA OPERATION)

    FOR SHORT = 1 TO 10000
    NEXT SHORT
    LOCATE 15, 20: PRINT " *** VALVE MOVING **** "
REM TRIGGER MOTOR CONTROLLER BY SENDING THE LETTER "A" OVER
SERIAL
REM COM #2 TO THE MOTOR PC.
    PRINT #2, "A"
REM
REM FOR AN IBM AT AND 0.01 SECOND INTERVALS SET TRSTEP = 80
REM TRSTEP WAS SET AT THE BEGINNING OF THIS PROGRAM.
REM
    FOR K = 1 TO POINTS STEP 1
    VOLTS = VSLOPE * POSTIME(K) + VCLOSED
    DIGITAL = INT(DSLOPE * VOLTS + DLOW)
        DH% = INT(DIGITAL / 16!)
        DL% = DIGITAL - 16! * DH%
        DL% = 16! * DL%
        OUT &H340, DL%
        OUT &H341, DH%
        FOR SLOW = 1 TO TRSTEP
        NEXT SLOW
    NEXT K
    OLDANGLE = POSTIME(POINTS)
    CLS
    LOCATE 10, 20: PRINT " *** FINISHED MOVING VALVE *****"

REM PUT A SHORT DELAY HERE TO ALLOW MASSCOMP COMPUTER TO
REM FINISH. THIS ALLOWS THE MASSCOMP TO STOP ACQUIRING DATA
REM WHEN THE CLOCK DICTATES (@ 5 SECONDS) RATHER THAN THE
REM TRIGGER

REM SET SHORT = 1 TO 10000 FOR NORMAL OPERATION
REM SET SHORT = 1 TO 9000 FOR CAMERA OPERATION WITH 5 SECOND
REM RUN AND 500 FPS FRAME RATE

    FOR SHORT = 1 TO 12000
    NEXT SHORT
    GOSUB TRIGLOW
    CLOSE #2
    GOTO 400

REM

```

```

REM *****
REM  TRANSIENT OPTION #5: CLOSE VALVE (SET TO 0 DEGREES)
REM *****
560  CLS

      NEWANGLE = 0!
      LOCATE 10, 20: PRINT " >>>>>> CLOSING VALVE <<<<<<<"
      GOSUB MOVEVALVE
      LOCATE 20, 20: PRINT " >>>>>> VALVE CLOSED <<<<<<<"
      FOR SHORT = 1 TO 12000
      NEXT SHORT
      GOTO 400

REM *****
600  PRINT " END OF PROGAM "
END
REM *****
REM
REM
REM          SUBROUTINES
REM
REM
REM *****
REM          STEADY STATE MOVEMENT OF VALVE
REM *****

```

MOVEVALVE:

```

      LOCATE 15, 20: PRINT " *****  MOVING VALVE *****"

      IF NEWANGLE = OLDANGLE THEN 800
      IF NEWANGLE < OLDANGLE THEN KANGLE = -STEPANGLE
      IF NEWANGLE > OLDANGLE THEN KANGLE = STEPANGLE

      FOR VALVEANGLE = OLDANGLE TO NEWANGLE STEP KANGLE
      VOLTS = VSLOPE * VALVEANGLE + VCLOSED
      DIGITAL = INT(DSLOPE * VOLTS + DLOW)
      DH% = INT(DIGITAL / 16!)
      DL% = DIGITAL - 16! * DH%
      DL% = 16! * DL%
      OUT &H340, DL%
      OUT &H341, DH%
      SHOWANGLE = INT(VALVEANGLE * 100) / 100!
      LOCATE 18, 20: PRINT " CURRENT VALVE POSITION : "; SHOWANGLE
      LOCATE 18, 55: PRINT "( degrees ) "
      FOR J = 1 TO STDSTEP

```

```
      NEXT J
      NEXT VALVEANGLE
800   OLDANGLE = NEWANGLE
```

```
RETURN
```

```
REM *****
```

```
REM          MASSCOMP HIGH TRIGGER
```

```
REM *****
```

```
TRIGHIGH:
```

```
    DL% = 0
```

```
    DH% = 125
```

```
    OUT &H342, DL%
```

```
    OUT &H343, DH%
```

```
    RETURN
```

```
REM *****
```

```
REM          MASSCOMP LOW TRIGGER
```

```
REM *****
```

```
TRIGLOW:
```

```
    DL% = 0
```

```
    DH% = 0
```

```
    OUT &H342, DL%
```

```
    OUT &H343, DH%
```

```
    RETURN
```

APPENDIX B: MOTOR CONTROLLER OPERATION

MOTOR CONTROLLER OPERATION (ON PC XT)

1. CHANGE TO BURTON DIRECTORY (CD \BURTON)
2. RUN "MOTORS" PROGRAM
3. CONTINUE AS BELOW FOR EITHER STEADY STATE OR
TRANSIENT OPERATION

STEADY STATE OPERATION

1. TYPE "P" FOR PARAMETER MENU
2. SET (WRITE)
#155 = 0 MOTOR RPM VIA POTENTIOMETER
= 1 MOTOR RPM VIA COMPUTER (MAKE SURE
POTENTIOMETER IS SET TO ZERO RPM)
3. IF USING COMPUTER CONTROL:
- SET RPM VIA #22 = APPROPRIATE NO. OF COUNTS
(0.41625 COUNTS/RPM)

TRANSIENT OPERATION (PARAMETER VALUES SUPPLIED FOR EXAMPLE ONLY)

1. TYPE "P" FOR PARAMETER MENU
2. SET (WRITE):
#155 = 1 (COMPUTER CONTROL)
55=60 (PROPORTIONAL GAIN)
56=40 (INTEGRAL GAIN)
3. ON MAIN MENU: TYPE "L" AND LOAD "RPM3.DAT" DATA FILE
4. ON MAIN MENU: TYPE "U" TO UPLOAD THE DATA FILE TO THE
CONTROLLER
5. AS DIRECTED ON SCREEN: HIT "ENTER" WHEN READY TO RUN
(PROGRAM WILL WAIT 60 SECONDS FOR TRIGGER FROM IBM AT)

NOTE: IF THE MENTOR CONTROLLER TRIPS AND THE RESET
BUTTON MUST BE PRESSED. THEN YOU MUST QUIT THE
MOTORS PROGRAM AND THEN RE-RUN IT IN ORDER TO
COMMUNICATE WITH MENTOR CONTROLLER.

MOTOR CONTROLLER PARAMETERS

22 RPM
55 PROPORTIONAL GAIN
56 INTEGRAL GAIN
99 =255; MENTOR RESET
150 =1; CHANGES DEFAULTS TO
CURRENT VALUES
(WAIT 3 MINUTES THEN RESET MENTOR)
166 =1; DISABLES THYRISTERS

COUNT SETTINGS FOR RESPECTIVE MOTOR SPEED (USED WITH PARAMETER NO. 22)

$$\frac{999 \text{ COUNTS}}{2400 \text{ RPM}} = 0.41625$$

RPM	COUNTS
300	125
500	208
600	250
750	312
900	375
1000	416
1200	500
1500	624
1800	749
2000	832
2400	999

**APPENDIX C: DEAERATION AND FILTRATION
PROCEDURES**

DEAERATION AND FILTRATION PROCEDURES FOR IMPELLER TEST FACILITY

1. Make sure the valves to the deaerator as well as filter supply and return lines from the Unsteady Flow Loop Facility are closed.
2. Remove the pressure to the Impeller Test Facility.
3. Open the valves in the supply and return lines to the deaerator and filter at the Impeller Test Facility.
4. Turn on the Tow Tank roof vents with switches EF6 and EF7 located in the boiler room.
5. Turn on deaerator circuit breakers on panel #17 located at the back wall of the Tow Tank area.
6. On the deaerator control panel located at the southeast corner of Building 1246 perform the following:
 - 6.1 Turn control power switch to "ON" position.
 - 6.2 Turn vacuum pump switch to "ON" position.

After the vacuum gauge reads approximately 20 inches Hg,

 - 6.3 Turn fill solenoid switch to "AUTO" position.
 - 6.4 Turn water pump switch to "AUTO" position.
7. Remain near the deaerator for several minutes to assure it is operating properly. Check the following:
 - 7.1 Water pump is switching on and off properly.
 - 7.2 Vacuum pump is holding proper vacuum.
- 8.0 Start filter pump by pressing green "ON" button located on building I-beam next to the Impeller Test Facility's instrumentation rack.
- 9.0 To shut down the filter and deaeration setup, shut off switches in the reverse order as at start up.

NOTE: Prior to every use of the deaerator check the sight glass on the vacuum pump to ensure there is enough lubricating oil for proper operation.

APPENDIX D: FACILITY OPERATION

IMPELLER TEST FACILITY OPERATIONAL PROCEDURES

- 1.0 Before facility startup check to assure that the following is in order:
 - 1.1 The main switch gear in the back machinery room must be set to the Impeller Test Facility position. This automatically disengages power to the Unsteady Flow Loop Facility and the Tow Tank carriage motors (when installed).
 - 1.2 Facility drains are closed. Note that there is a drain at the bottom of the test section and another at the bottom of the plenum.
 - 1.3 Manometer lines leading to the cavitation susceptibility meter are closed. The meter can be damaged if pressure is applied.
 - 1.4 The facility must be full of water.
 - 1.5 The facility water supply valve located along the back wall of the building must be closed.
 - 1.6 Valves in all lines leading to auxiliary equipment (deaerator, filter, etc.) must be closed.
 - 1.7 Motor controller cabinet doors are closed and safety interlock is activated. Note that the lock on the door must be removed and the lever set to the "ON" position. Make sure that the reset button on the motor controller has been pressed and is no longer lit. Press the "START" button on the motor control above the potentiometer control.
 - 1.8 Start the hydraulic pump and check that fluid temperature is no greater than 150 degrees Fahrenheit, and its pressure is no greater than 3000 psig. Should the temperature increase to the stated level, the cooling water can be circulated through the pump's cooling coils.
 - 1.9 Facility static (depth) pressure is never to exceed 100 psig immediately prior to impeller testing. With the stainless steel inlet pipe in place of the acrylic one, 200 psig static pressure can be applied for sensor calibrations without the impeller operating; the clear acrylic inlet pipe cannot be subjected to pressures in excess of 100 psig. If the pressure was raised above 100 psig for calibration purposes, it must be lowered to a maximum of 100 psig as soon as possible after calibration is completed.
 - 1.10 The safety railing at the motors is to be installed at all times while the motors are operating.

- 1.11 If computer control of the motors is to be used, the manual speed adjustment potentiometer located near the motor cabinet must be set to zero.
- 1.12 Ensure that there are no personnel near the vicinity of the rotating motor and impeller shafts.

2.0 Preliminary Facility Startup and Miscellaneous Sensor Calibration

- 2.1 Check to ensure all valves to auxiliary equipment have been closed as stated in section 1.6 above.
- 2.2 Power up all instrumentation and computers, which includes plugging the Hiese pressure gauge into the 24-volt control valve power supply.
- 2.3 Plug the transient flowmeter body into the power strip located at the base of the hydraulic power unit.
- 2.4 Allow the instrumentation and computer to warm up for 15 minutes.
- 2.5 Open the filter line located behind the motor input shaft along with the facility water supply line to pressure the facility to tap pressure (approximately 40 psig). Then close the supply line. The filter line may remain open.
- 2.6 Load the MOTORS.BAS program onto the PC XT as described in Appendix B.
- 2.7 Assure that the control valve is open to 90 degrees. (See items 2.13 and 2.14 if valve is not at 90 degrees open.)
- 2.8 Ramp up the motors manually to 1200 RPM in 150 RPM steps and allow to run until the facility is purged of any trapped air. Shut down the
motors and re-pressurize the facility to tap pressure. Repeat this step several times if necessary.
- 2.9 Set zero on both the NUWC and Foxboro electromagnetic flowmeter electronics, then calibrate the NUWC flowmeter as follows:
 - 2.9.1 Ramp up the motors in steps to obtain the maximum flowrate.
 - 2.9.2 Adjust the span on the NUWC electronics so that the voltage output is 0.8 times that of the Foxboro electronics when set to the "B" range.

- 2.9.3 Shut down the motors.
- 2.10 Open the valve leading to the Heise pressure gauge.
- 2.11 Set zero on the Ectron amplifiers used for the Kulite pressure transducers with the facility pressure regulated to a value of -0.136 volts on the Heise pressure gauge. The balance screw adjustment on the amplifier is used.
- 2.12 Calibrate the pressure transducers to the facility's maximum operating pressure as follows:
 - 2.12.1 Open the facility water supply and pressurize the facility to tap pressure. Close the water supply line and filter line at back wall.
 - 2.12.2 Open the 100 psi air supply valve, located on the I-beam next to the facility's instrumentation rack, which leads to the SC Hydraulic Engineering hydro pump located under the plenum of the facility. Ensure the pump's air pressure is regulated to zero.
 - 2.12.3 Connect the water supply of the hydro pump to the sink faucet located behind the Small Launcher Facility (SLF) control room.
 - 2.12.4 Open the valve on the facility pressurization line from the hydro pump to the facility, which is located on the upper elbow above the impulse tank.
 - 2.12.5 Prime the hydro pump by opening the water isolation valve on the hydro pump water inlet.
 - 2.12.6 Regulate the hydro pump's air pressure to approximately 60 psig.
 - 2.12.7 Open air shut-off valve to begin pressurization of the facility.
 - 2.12.8 Pressurize the facility to 90 psig, which corresponds to 8.864 volts on the Heise pressure gauge.
 - 2.12.9 Shut down the hydro pump by closing the air shut-off valve.
 - 2.12.10 Close the valve on the facility pressurization line located on the upper elbow of the facility above the impulse tank.
 - 2.12.11 Use the vernier screw adjustments on the Ectron amplifiers to set pressure transducer voltage to 2.25 volts, which corresponds to 8.864 volts on the Heise gauge. This 0-to-2.25-volt range ensures a linear relationship of 40 psi per volt on the pressure transducers.

- 2.12.12 Relieve the facility pressure so that a voltage of -0.136 volts is observed on the Heise gauge.
- 2.12.13 Set the zero on the pressure transducers once again as done in 2.11 above.
- 2.12.14 Close the valve leading to the Heise gauge.
- 2.13 On the IBM AT computer load the program VCONIMPL.BAS located in the C:\CONTROL\13SCALE directory of the computer.
- 2.14 For steady-state operation perform the following steps:
 - 2.14.1 Turn on the power supply to the hydraulic power unit by throwing the switch located on the front of the power unit control box .
 - 2.14.2 Power up the hydraulic power unit by pressing the "ON" button located on the control box of the power unit.
 - 2.14.3 Activate the 16-inch control valve actuated by the hydraulic power unit by placing the switch on the servo-valve control box, which is located on the shelf directly below the IBM AT, to the "ON" position.
 - 2.14.3 Regulate the hydraulic power unit's pressure to 1000 psig using the yellow valve located on the side of the power unit directly below the unit's pressure gauge.

NOTE: Do not leave the hydraulic pump (power unit) operating at higher pressure for extended durations since over-heating may occur.
 - 2.14.4 Select the "Steady-state valve operation" selection from the main menu of the VCONIMPL.BAS program.
 - 2.14.5 Enter the desired valve angle. The control valve will move to the selected position.
- 2.15 For transient operation of the facility perform the following:
 - 2.15.1 Activate the control valve as in section 2.13, but set the hydraulic pressure to 2500 to 3000 psi.

- 2.15.2 Shut down the hydraulic power unit using the servo valve control box. Turn the power unit back on to hold the valve at the desired position.
- 2.15.3 On the main menu of the VCONIMPL.BAS select the "Transient valve operation" selection.
- 2.15.4 From the "Transient Valve Operation" menu select "Get valve position data from file".
- 2.15.5 Enter the name of the valve position versus time file to be used for testing (e.g., DEG3.DAT).
- 2.15.6 Select "Run existing data" to load the selected file into memory. Upon activation of the hydraulic pump, the control valve will close.
- 2.15.7 On the PC XT computer load the RPM versus time file, RPM3.DAT into the computer as described in Appendix B of this report.

3.0 Test Procedures

- 3.1 Pressurize the facility to the desired pressure. If the desired pressure is less than tap pressure (40 psig) the facility water supply may be used as in section 2.5. If test pressure is greater than tap pressure, the hydro pump must be used. Follow the procedures in sections 2.11.1 to 2.11.10. The test pressure may be set exactly by relieving pressure using the drain at the bottom of the plenum.
- 3.2 Ensure all valves leading to auxiliary equipment are closed.
- 3.3 Adjust the gain settings on the Ithaco charge amplifiers used to gain the accelerometers to proper level for the test pressure to be used. Note that since these amplifiers are charge instead of voltage amplifiers, an inverse relationship exists between the gain setting and the actual signal gain (i.e., the higher the gain setting the less the signal is amplified).
- 3.4 Turn on the magnetic tape recorder and record the following voice information: date, test run name and number, test pressure, water temperature, accelerometer tape channels and gain settings, and any other pertinent information.
- 3.5 Activate the data acquisition system by the following procedure:

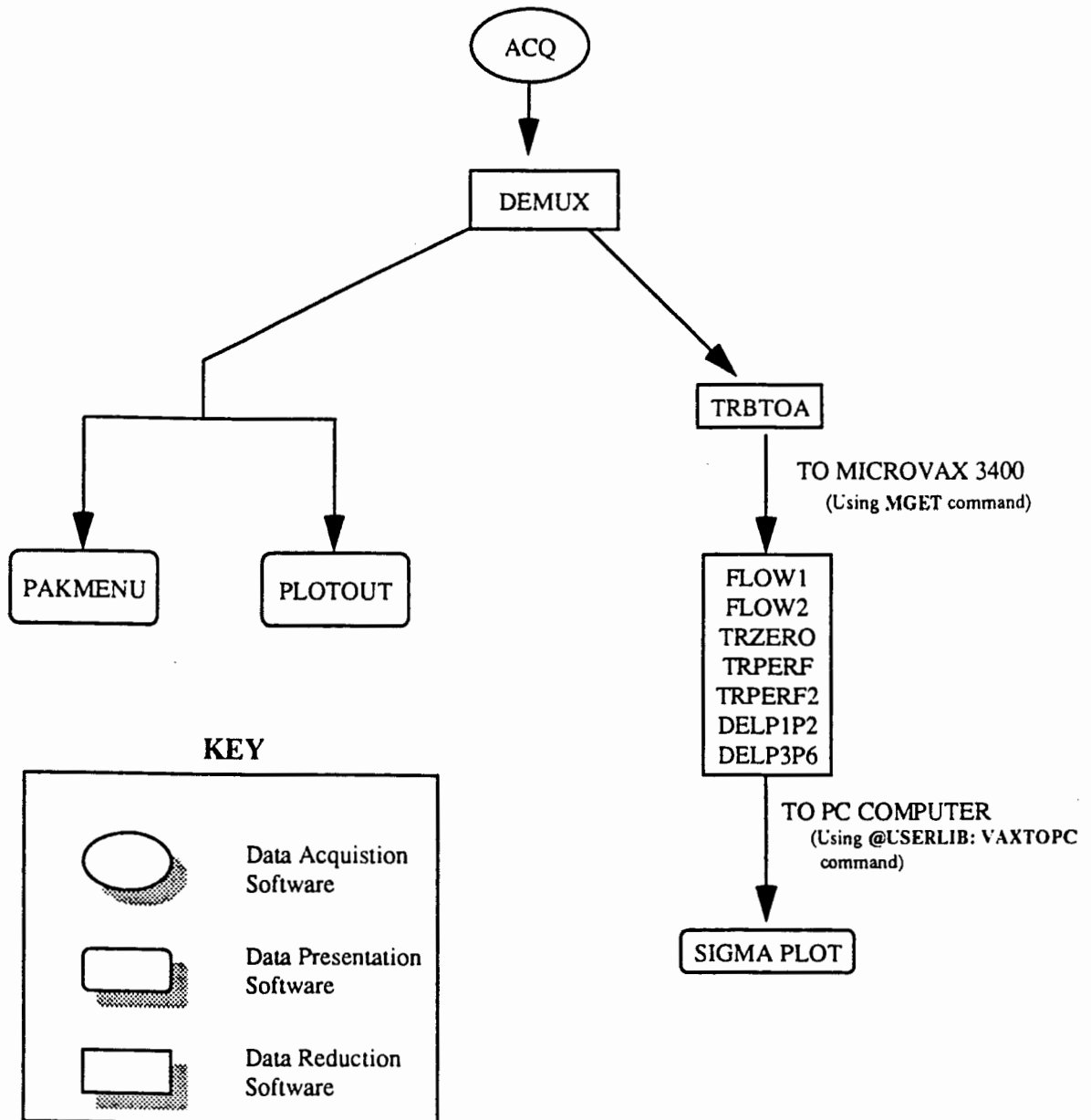
- 3.5.1 Log in to the IMPELLER directory of the MASSCOMP 6600 using the DEC VT220 terminal located on the computer bench.
- 3.5.2 Enter the \PROG\BILL directory and type "acq" to run the ACQ.C data acquisition program.
- 3.5.3 Enter the following information as prompted by the program: test name, random or sequential channel sampling, number of channels to be sampled, channel numbers to be sampled, sampling frequency, duration of data acquisition, and time delay to initiate data acquisition.
- 3.5.4 The program will prompt the operator to type "GO" and hit the return key to put the MASSCOMP into a 60-second wait state for data acquisition. Type "GO" but do not hit return unless the trigger signal from the IBM AT within the 60 seconds.
- 3.6 If the "Run existing data" selection on the IBM AT has not been chosen, as in section 2.14.6, do so. The screen will display a message that the control valve is closing along with a display of the changing valve angle.
- 3.7 When the AT machine's monitor displays the message "Hit return key to start transient valve movement", turn on the hydraulic power unit (pump) as done in 2.15.2.
- 3.8 Upload the selected RPM versus time file into the XT machine as described in Appendix B. Hit the any key to send the motor controller into a wait state for the trigger signal from the AT computer.
- 3.9 Depress the return key on the VT220 terminal to place the MASSCOMP data acquisition system into a wait state for the trigger signal.
- 3.10 Activate the magnet tape recorder, start an audible countdown, and upon end of countdown depress the return key on the AT computer to initiate the test.
- 3.11 For steady-state testing the motors are controlled through the XT by the test personnel instead of a data file. To acquire data under steady-state conditions, only triggering of the MASSCOMP is required.
- 3.12 After the VT220 terminal displays the message that the data have indeed been acquired, shut down the hydraulic pump.

4.0 Facility Shut Down Procedures

- 4.1 Remove the pressure from the facility using the facility drains.
- 4.2 Disconnect the electricity to the transient flowmeter.
- 4.3 Make sure that all lines (air and water) leading to and from the hydro pump are shutoff and depressurized.
- 4.4 Shut down power to all instrumentation and computers.
- 4.5 Shut down and lock the motor controller cabinet as well as the hydraulic pump control box by moving switches to "OFF" position. Make sure to push the stop button on the motor controller before switching off the cabinet.

**APPENDIX E: DATA ACQUISITION, REDUCTION, AND
PRESENTATION FLOWCHART**

Data Acquisition, Reduction, and Presentation Flowchart



**APPENDIX F: LASER SAFETY PROCEDURES
FOR BUILDING 1246**

1.1 INTRODUCTION:

This document puts forth the safety precautions and standard operating procedures required for the safe operation of lasers located in Building 1246.

2.1 BACKGROUND:

The Flow Loop Facility and the Impeller Test Facility, which are located in building 1246 of NUWC's Newport Laboratory, both use Laser Doppler Velocimetry (LDV) as an experimental tool for flow studies. These facilities also use Particle Image Velocimetry (PIV) as a means of obtaining flow visualization data. The principal component used in both these techniques is a laser. As such, appropriate safety precautions and operating procedures must be adhered to at all times to ensure both safe and proper use of these facilities.

3.1 OBJECTIVE:

The objective of this document is to establish safety guidelines and standard operating procedures for the use of the three lasers used in building 1246.

4.1 LASER SAFETY DOCUMENTS

The following is a list of documents pertaining to the Laser Safety Program:

4.1.1 ANSI Z136.1 - Published by American National Standards Institute, Inc., has been adopted as the Navy document for the Laser Safety Program.

4.1.2 SPAWARINST 5100.12A - All safety and operational requirements of this document apply to the use of the lasers located in building 1246.

5.1 LIST OF HAZARDS:

5.1.1 Exposure to laser light can result in tissue damage, primarily burns. The eyes, which focus light, are the most vulnerable to the effects of laser light. The maximum permissible exposure levels for both skin and eyes can be found in the reference listed in section 4.1.1.

5.1.2 The high power (class IV) lasers used are run on high voltage, which may present a problem either alone or coupled with the fact that these lasers also are water cooled.

6.1 LASER DESCRIPTIONS:

6.1.1 Laser 1: Uniphase Helium - Neon Laser Model 105-1

Power: 5 mW
Class: IIIB
Wavelength: 632.8 nm (Red)
Use: Camera light source
Location: Data analysis room located on east wall of Bldg. 1246.

6.1.2 Laser 2: CJ Laser Copper - Vapor Series 2000

Power: 25 watts total, 12.5 watts per wavelength
Class: IV
Wavelengths: 511 nm (Yellow), 578 nm (Yellow-green)
Use: Particle Illumination
Location: Flow Loop Facility Laser Room, Bldg. 1246.

6.1.3 Laser 3: Lexel Argon - Ion Laser Model 95-4

Power: 2 watts
Class: IV
Wavelengths:

<u>Wavelength (nm)</u>	<u>Power (watts)</u>
457.9	0.16
476.5	0.24
488.0	0.40
514.5	0.86
520.7	0.20

Use: Laser Doppler Velocimetry
Location: Flow Loop Facility Laser Room, Bldg. 1246.

7.1 SAFETY PROCEDURES:

7.1.1 In case of emergency lasers can be shut down with laser main power breakers located in the breaker box at the southeast corner of building 1246.

7.1.2 Only NUWC personnel who have met the medical and training requirements and have been qualified by the designated Laser Systems Safety Officer shall be authorized to operate the laser systems.

7.1.3 Entrance to the areas of laser use shall only be with the permission of authorized personnel.

7.1.4 Spectators shall not be permitted in the lab without appropriate eye protection as dictated by the laser to be viewed.

7.1.5 Reflective articles of personnel attire shall not be worn during operations in which they may cause hazardous reflections of the laser beams.

7.1.6 Lasers shall not be left unattended while operating.

7.1.7 Personnel will adhere to NUSCINST 5100.15 Lockout/tagout instruction as appropriate for the troubleshooting of laser systems.

7.1.8 The warning beacon must be in operation, and all the exposed areas will be cleared of all personnel except those running the laser and those observers permitted with appropriate bodily protection (eye, etc.), and also a verbal warning must be given before laser operations may commence.

7.1.9 During laser operation, the door interlocks shall be engaged to ensure that the laser will automatically shut down if the entrance to the flow loop facility is opened.

7.1.10 When the fiber optic cable and probe is utilized to transmit the laser beam to a remote location, partitions shall be installed around the laser beam working area. The partitions shall have a height that precludes anyone outside the partitioned area from having direct sight of the beam or its reflections. Signs will be affixed to these partitions to warn personnel of the presence of the laser beam. The entrance to the partitioned area can be open (no door) but must be located in a manner that eliminates any inadvertent exposure to the beam and its reflections to other personnel. A warning beacon is to be installed near the entrance.

7.1.11 When aiming the laser, avoid looking along the axis of the beam to avoid accidental contact with reflected beams.

7.1.12 Protective gloves, shielding, clothing, etc. that are resistant to hazardous laser energy will be used when deemed necessary.

7.1.13 Suitable warning signs will be posted in areas where lasers are operated.

7.1.14 Prior to operating the laser, inspection of the immediate area should be conducted to ensure there are no reflective surfaces or they have been properly covered.

8.1 OPERATING PROCEDURES

8.1.1 When the laser system is not in-use, the main power switch-disconnect shall be in the locked OFF position.

8.1.2 This system shall never be configured to operate in a manner which permits the laser beam to exit from the flow visualization section of the facility in question.

8.1.3 Operating and safety procedures and recommendations of the laser manufacturer should be observed.

8.1.4 Beam alignment shall be performed at the minimum power level necessary for such a task.

8.1.5 A copy of laser operational procedures from the manufacturer as well as those from the Navy shall be kept with lasers.

DISTRIBUTION LIST

External

CNR (OCNR-1132F (P. Purtell), OCNR-12 (J. Fein), OCNR-23, OCNR-30 (J. DeCorpo),
ONT23 (G. Remers))

DTRC (Codes 0114 (L. Becker), 15 (Morgan), 1543 (D. Coder), 1843 (H. Haussling),
2714 (J. Henry), 2741 (T. Calvert))

NAVSEA (SEA-56W, SEA-56W13, PMS-350C)

Internal

Codes	0212
	0261
	0262 (2)
	02244
	10
	102 (K. Lima)
	22
	38
	3891
	80
	804
	804 (J. Uhlman)
	8093
	81
	821
	83
	8302
	832
	8321 (W. Barker (2), M. Kiss, W. McInnis, M. Little, M. Ansay)
	8322 (P. Lefebvre (3), R. Gregory (5), M. Rodrigues (2))
	833

Total: 50 Copies